

**Министерство образования и науки Российской Федерации**  
федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

---

Школа – Информационных технологий и робототехники  
Направление подготовки – 09.04.01 Информатика и вычислительная техника  
Отделение – Информационных технологий

**МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ**

Тема работы
<b>Разработка алгоритма автоматического управления роботизированными механизмами с применением искусственных нейронных сетей</b>

УДК 004.421:621.865.8:004.7.032.26

Студент

Группа	ФИО	Подпись	Дата
8BM93	Пантюхин Евгений Игорьевич		

Руководитель

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Иванова Ю.А.	К.Т.Н.		

**КОНСУЛЬТАНТЫ:**

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН	Верховская М.В.	К.Э.Н.		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Профессор ООД ШБИП	Федоренко О.Ю.	Д.М.Н		

**ДОПУСТИТЬ К ЗАЩИТЕ:**

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Профессор ОИТ ИШИТР	Спицын В.Г.	Д.Т.Н.		

## ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

по направлению 09.04.01 Информатика и вычислительная техника

<b>Код компетенции</b>	<b>Наименование компетенции</b>
<b>Универсальные компетенции</b>	
УК(У)-1	Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, вырабатывать стратегию действий
УК(У)-2	Способен управлять проектом на всех этапах его жизненного цикла
УК(У)-3	Способен организовывать и руководить работой команды, вырабатывая командную стратегию для достижения поставленной цели
УК(У)-4	Способен применять современные коммуникативные технологии, в том числе на иностранном (-ых) языке (-ах), для академического и профессионального взаимодействия
УК(У)-5	Способен анализировать и учитывать разнообразие культур в процессе межкультурного взаимодействия
УК(У)-6	Способен определять и реализовывать приоритеты собственной деятельности и способы ее совершенствования на основе самооценки
<b>Общепрофессиональные компетенции</b>	
ОПК(У)-1	Способен самостоятельно приобретать, развивать и применять математические, естественно-научные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте
ОПК(У)-2	Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач
ОПК(У)-3	Способен анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров с обоснованными выводами и рекомендациями
ОПК(У)-4	Способен применять на практике новые научные принципы и методы исследований
ОПК(У)-5	Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем
ОПК(У)-6	Способен разрабатывать компоненты программно-аппаратных комплексов обработки информации и автоматизированного проектирования

ОПК(У)-7	Способен адаптировать зарубежные комплексы обработки информации и автоматизированного проектирования к нуждам отечественных предприятий
ОПК(У)-8	Способен осуществлять эффективное управление разработкой программных средств и проектов
<b>Профессиональные компетенции</b>	
ПК(У)-1	Способен разрабатывать и администрировать системы управления базами данных
ПК(У)-2	Способен проектировать сложные пользовательские интерфейсы
ПК(У)-3	Способен управлять процессами и проектами по созданию (модификации) информационных ресурсов
ПК(У)-4	Способен осуществлять руководство разработкой комплексных проектов на всех стадиях и этапах выполнения работ
ПК(У)-5	Способен проектировать и организовывать учебный процесс по образовательным программам с использованием современных образовательных технологий

**Министерство образования и науки Российской Федерации**  
федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Школа – Информационных технологий и робототехники  
Направление подготовки (специальность) – 09.04.01 Информатика и вычислительная техника  
Отделение – Информационных технологий

УТВЕРЖДАЮ:  
Зав. отделением

\_\_\_\_\_  
(Подпись)      (Дата)      (Ф.И.О.)

**ЗАДАНИЕ**  
**на выполнение выпускной квалификационной работы**

В форме:

Магистерская диссертация
--------------------------

(бакалаврской работы, дипломного проекта/работы, магистерской диссертации)

Студенту:

Группа	ФИО
8BM93	Пантюхин Евгений Игорьевич

Тема работы:

Разработка алгоритма автоматического управления роботизированными механизмами с применением искусственных нейронных сетей	
Утверждена приказом директора (дата, номер)	

Срок сдачи студентом выполненной работы:	
--	--

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ:**

<p><b>Исходные данные к работе</b></p> <p><i>(наименование объекта исследования или проектирования; производительность или нагрузка; режим работы (непрерывный, периодический, циклический и т. д.); вид сырья или материал изделия; требования к продукту, изделию или процессу; особые требования к особенностям функционирования (эксплуатации) объекта или изделия в плане безопасности эксплуатации, влияния на окружающую среду, энергозатратам; экономический анализ и т. д.).</i></p>	<p>Гаврилов А.В., Губарев В.В., Джо К.Х., Ли Х.Х., Архитектура гибридной системы управления мобильного робота/ Научный вестник НГТУ, 2004, №2</p> <p>Хайкин, Нейронные сети: полный курс. 2 издание. С. Хайкин. – Москва: Издательский дом "Вильямс", 2006. – 1104 с.</p>
---	---

<p><b>Перечень подлежащих исследованию, проектированию и разработке вопросов</b></p> <p><i>(аналитический обзор по литературным источникам с целью выяснения достижений мировой науки техники в рассматриваемой области; постановка задачи исследования, проектирования, конструирования; содержание процедуры исследования, проектирования, конструирования; обсуждение результатов выполненной работы; наименование дополнительных разделов, подлежащих разработке; заключение по работе).</i></p>	<ol style="list-style-type: none"> <li>1. Исследовать структуру системы управления автономными аппаратами.</li> <li>2. Изучить существующие методы обучения нейронных сетей.</li> <li>3. Исследовать особенности обучения ИНС по управлению роботизированными механизмами.</li> <li>4. Разработать программные средства для моделирования работы манипулятора на основе ИНС.</li> <li>5. Провести многоступенчатое тестирование и анализ полученных результатов.</li> </ol>
<p><b>Перечень графического материала</b></p> <p><i>(с точным указанием обязательных чертежей)</i></p>	<p>29 рисунков, 16 таблиц, презентация Microsoft PowerPoint</p>
<p><b>Консультанты по разделам выпускной квалификационной работы</b></p> <p><i>(с указанием разделов)</i></p>	
<p><b>Раздел</b></p>	<p><b>Консультант</b></p>
<p><b>Финансовый менеджмент, ресурсоэффективность и ресурсосбережения</b></p>	<p>Доцент ОСГН Верховская Марина Витальевна</p>
<p><b>Социальная ответственность</b></p>	<p>Профессор ООД ШБИП Федоренко Ольга Юрьевна</p>
<p><b>Приложение А</b></p>	<p>Розанова Яна Викторовна</p>
<p><b>Названия разделов, которые должны быть написаны на русском и иностранном языках:</b></p>	
<p>Введение</p> <p>Структура системы управления роботизированными автономными аппаратами</p>	

<p><b>Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику</b></p>	
--	--

**Задание выдал руководитель:**

<p>Должность</p>	<p>ФИО</p>	<p>Ученая степень, звание</p>	<p>Подпись</p>	<p>Дата</p>
<p>Доцент</p>	<p>Иванова Ю.А.</p>	<p>к.т.н.</p>		

**Задание принял к исполнению студент:**

<p>Группа</p>	<p>ФИО</p>	<p>Подпись</p>	<p>Дата</p>
<p>8BM93</p>	<p>Пантюхин Е.И.</p>		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА  
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И  
РЕСУРСОСБЕРЕЖЕНИЕ»**

Студенту:

<b>Группа</b>	<b>ФИО</b>
8BM93	Пантюхину Евгению Игоревичу

<b>Школа</b>	<b>ИШИТР</b>	<b>Отделение школы (НОЦ)</b>	<b>ОИТ</b>
Уровень образования	магистратура	Направление/специальность	09.04.01 «Информатика и вычислительная техника»

**Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:**

1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих	Затраты на ЗП: Инженер – 124 866 руб. Руководителя – 18 689 руб.
2. Нормы и нормативы расходования ресурсов	Районный коэффициент 30%; Коэффициент дополнительной заработной платы 15%; Накладные расходы 16%.
3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования	Коэффициент отчислений на уплату во внебюджетные фонды 30%

**Перечень вопросов, подлежащих исследованию, проектированию и разработке:**

1. Оценка коммерческого потенциала, перспективности и альтернатив проведения НИ с позиции ресурсоэффективности и ресурсосбережения	-Анализ конкурентных технических решений
2. Планирование и формирование бюджета научных исследований	Формирование плана и графика разработки: - определение структуры работ; - определение трудоемкости работ; - разработка графика Ганта. Формирование бюджета затрат на научное исследование: - затраты на специальное оборудование; - заработная плата (основная и дополнительная); - отчисления на социальные цели; - накладные расходы.
3. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования	- Определение потенциального эффекта исследования

**Перечень графического материала (с точным указанием обязательных чертежей):**

1. Оценочная карта конкурентных технических решений
2. Матрица SWOT
3. График Ганта
4. Расчет бюджета затрат

<b>Дата выдачи задания для раздела по линейному графику</b>	
---	--

**Задание выдал консультант:**

<b>Должность</b>	<b>ФИО</b>	<b>Ученая степень, звание</b>	<b>Подпись</b>	<b>Дата</b>
Доцент ОСГН	Верховская М.В.			

**Задание принял к исполнению студент:**

<b>Группа</b>	<b>ФИО</b>	<b>Подпись</b>	<b>Дата</b>
8BM93	Пантюхин Е.И.		

## ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
8BM93	Пантюхин Евгений Игорьевич

Школа	ИШИТР	Отделение (НОЦ)	ОИТ
Уровень образования	Магистратура	Направление/специальность	09.04.01 Информатика и вычислительная техника

Тема ВКР:

Разработка алгоритма автоматического управления роботизированными механизмами с применением искусственных нейронных сетей	
Исходные данные к разделу «Социальная ответственность»:	
1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области его применения	<p><i>Объект исследования - технология по управлению роботизированными механизмами. Область применения – любые высокотехнологичные автономные аппараты, способные взаимодействовать с окружающей средой.</i></p> <p>– Рабочее помещение представлено исследовательской лабораторией со следующими параметрами:</p> <ul style="list-style-type: none"> <li>– Общая площадь: 18м<sup>2</sup>;</li> <li>– Отопление: центральное отопление (радиатор на 4 секции);</li> <li>– Вентиляция: вытяжная вентиляция 20см<sup>2</sup>;</li> <li>– Освещение: 3 светильника;</li> </ul> <p>Оборудование: 2 персональных компьютера, роутер пр. периферийные устройства.</p>
Перечень вопросов, подлежащих исследованию, проектированию и разработке:	
<p><b>1. Правовые и организационные вопросы обеспечения безопасности:</b></p> <ul style="list-style-type: none"> <li>– специальные (характерные при эксплуатации объекта исследования, проектируемой рабочей зоны) правовые нормы трудового законодательства;</li> <li>– организационные мероприятия при компоновке рабочей зоны.</li> </ul>	<p>Трудовой кодекс Российской Федерации от 30.12.2001 N 197-ФЗ (ред. от 09.03.2021)</p> <p>СанПиН 2.2.4.548-96 Гигиенические требования к микроклимату производственных помещений.</p> <p>ГОСТ 12.2.032-78 Система стандартов безопасности труда (ССБТ). Рабочее место при выполнении работ сидя. Общие эргономические требования.</p> <p>ГОСТ Р ИСО 9241-4-2009 Эргономические требования к проведению офисных работ с</p>

	<p>использованием видеодисплейных терминалов (VDT)</p> <p>СанПиН 1.2.3685-21 Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания</p> <p>ГОСТ 12.0.003-2015 Опасные и вредные производственные факторы. Классификация. Перечень опасных и вредных факторов</p> <p>СП 52.13330.2016 Естественное и искусственное освещение. Актуализированная редакция</p> <p>СН 2.2.4/2.1.8.562-96. Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории жилой застройки</p> <p>ГОСТ 12.1.003-2014 ССБТ. Шум. Общие требования безопасности</p> <p>ГОСТ 12.1.003-83 Система стандартов безопасности труда (ССБТ). Шум. Общие требования безопасности</p> <p>ТОИ Р-45-084-01 Типовая инструкция по охране труда при работе на персональном компьютере</p> <p>ГОСТ 12.1.006-84 ССБТ «Электромагнитные поля радиочастот. Допустимые уровни на рабочих местах и требования к проведению контроля»</p> <p>ГОСТ 12.1.038-82 ССБТ</p> <p>ГОСТ 12.1.004-91 Система стандартов безопасности труда (ССБТ). Пожарная безопасность. Общие требования</p>
<p><b>2. Производственная безопасность:</b></p> <p>2.1. Анализ выявленных вредных и опасных факторов</p> <p>2.2. Обоснование мероприятий по снижению воздействия</p>	<p>Вредные факторы:</p> <ul style="list-style-type: none"> <li>– Недостаточная освещенность рабочего места;</li> <li>– Отклонение показателей микроклимата от нормы;</li> <li>– Превышение уровня шума;</li> <li>– Вибрации;</li> <li>– Психофизиологические факторы (монотонность труда, нервно-психические перегрузки, перенапряжение зрительных анализаторов);</li> </ul> <p>Опасные факторы:</p> <ul style="list-style-type: none"> <li>– Статическое электричество;</li> <li>– Поражение электрическим током;</li> <li>– Короткое замыкание</li> </ul>



<b>3. Экологическая безопасность:</b>	Негативное воздействие на литосферу происходит при утилизации компьютера и периферийных устройств (принтеры, МФУ, веб-камеры, наушники, колонки, телефоны), аккумуляторных батареек, люминесцентных ламп, макулатуры.
<b>4. Безопасность в чрезвычайных ситуациях:</b>	Вероятные ЧС: пожары, грозы, ураганы, оползни. Наиболее типичные ЧС: Пожар.

<b>Дата выдачи задания для раздела по линейному графику</b>	01.03.2021
---	------------

**Задание выдал консультант:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Профессор ООД ШБИП	Федоренко О.Ю.	д.м.н		01.03.2021

**Задание принял к исполнению студент:**

Группа	ФИО	Подпись	Дата
8ВМ93	Пантюхин Е.И.		01.03.2021

**Министерство образования и науки Российской Федерации**  
федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Школа – Информационных технологий и робототехники  
Направление подготовки – 09.04.01 Информатика и вычислительная техника  
Уровень образования – Магистр  
Отделение – Информационных технологий  
Период выполнения – (осенний / весенний семестр 2020/2021 учебного года)

Форма представления работы:

Магистерская диссертация
--------------------------

(бакалаврская работа, дипломный проект/работа, магистерская диссертация)

**КАЛЕНДАРНЫЙ РЕЙТИНГ-ПЛАН**  
**выполнения выпускной квалификационной работы**

Срок сдачи студентом выполненной работы:	
--	--

Дата контроля	Название раздела (модуля) / вид работы (исследования)	Максимальный балл раздела (модуля)
15.01.21	<i>Раздел 1. Структура системы управления автономным аппаратом</i>	5 баллов
25.01.21	<i>Раздел 2. Искусственные нейронные сети</i>	5 баллов
20.02.21	<i>Раздел 3. Разработать программные средства для моделирования работы манипулятора на основе ИНС.</i>	10 баллов
15.04.21	Многоступенчатое тестирование и анализ полученных результатов. Оформление пояснительной записки.	10 баллов
15.05.21	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	10 баллов
15.05.21	Социальная ответственность	10 баллов
25.05.21	Раздел ВКР, выполненный на иностранном языке	10 баллов

Составил преподаватель:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент	Иванова Ю.А.	К.Т.Н.		

**СОГЛАСОВАНО:**

Руководитель ООП	ФИО	Ученая степень, звание	Подпись	Дата
Профессор ОИТ ИШИТР	Спицын В.Г.	Д.Т.Н.		

## РЕФЕРАТ

Выпускная квалификационная работа содержит 114 с., 29 рис., 16 табл., 37 источников, 1 прил.

Ключевые слова: нейронные сети, автономные аппараты, манипулятор, обучение с подкреплением, моделирование

Объектом исследования и разработки являются нейросетевые модели, предназначенные для управления автономными роботами.

Целью работы является разработка нейросетевого алгоритма управления манипулятором в смоделированной среде.

Методологией проведения работы является графическое моделирование физической среды, исследование методов обучения с подкреплением, программная реализация выбранного метода и проведение численных экспериментов по обучению модели роботизированных устройств.

Результатом проведённых работ является смоделированная среда для обучения интеллектуальных агентов выполнению задачи по захвату и поднятию объектов. Для апробирования данной разработки в реальных устройствах, необходимо сначала применить любые методы или алгоритмы по построению реконструкций окружения в виртуальной среде. После калибровки и обучения, ИНС будут способны выполнять поставленные задачи в реальном мире.

Результаты данной работы могут быть использованы для обучения реальных роботизированных устройств, способных взаимодействовать с окружающей средой посредством различных хватательных механизмов.

Значимость работы характеризуется тем, что разработанная концепция обучения роботизированных механизмов может существенно расширить спектр задач, решаемых автономными аппаратами;

В перспективе, на основе полученных знаний можно проектировать более сложные роботизированные системы, выполняющие задачи по автоматической сортировке и транспортировке объектов, проводящие

ассистирование на производстве, в лабораториях или в быту, выполняющие другие нешаблонные манипуляции с объектами.

## **ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ, СОКРАЩЕНИЯ, НОРМАТИВНЫЕ ССЫЛКИ**

В данной работе применены следующие термины с определениями:

1. ИНС – искусственная нейронная сеть;
2. PnP – Perspective-n-Point;
3. SIFT – scale-invariant feature transform;
4. SURF – speed up robust feature;
5. SAC – Soft Actor-Critic;
6. PPO – Proximal Policy Optimization;
7. Q-learning – метод, применяемый в искусственном интеллекте для обучения интеллектуальных агентов;
8. V-REP – симулятор для программирования роботов;
9. PhysX – физический движок для симуляции ряда физических явлений, а также комплект средств разработки на его основе;
10. ML-Agents – проект с открытым исходным кодом, который позволяет играм и симуляторам служить средой для обучения интеллектуальных агентов;
11. PyTorch – фреймворк машинного обучения для языка Python с открытым исходным кодом.

## СОДЕРЖАНИЕ

Введение.....	16
1. Структура системы управления автономными роботизированными механизмами.....	18
1.1. Система управления автономным аппаратом.....	18
1.2. Обучение алгоритмов в смоделированной среде .....	19
1.3. Рандомизация смоделированных данных .....	20
1.4. Локализация и классификация объектов.....	22
1.4.1. Локализация и классификация с применением карт глубины .....	23
1.4.2. Локализация и классификация без применения карт глубины .....	25
1.5. Архитектура гибридной системы управления .....	28
2. Общие принципы работы искусственных нейронных сетей.....	30
2.1. Общая информация .....	30
2.2. Принцип работы искусственных нейронных сетей .....	32
2.3. Парадигма обучения с учителем.....	34
2.4. Парадигма обучения без учителя.....	36
2.5. Парадигма обучения с подкреплением .....	37
2.5.1. Общая информация.....	37
2.5.2. Фундаментальные методы решения.....	39
2.5.3. Функция ценности.....	41
3. Разработка программных средств для моделирования работы манипулятора.....	43
3.1. Выбор средств реализации .....	43
3.2. Определение требований к проектируемой системе .....	45
3.3. Структура приложения .....	46
3.3.1. Класс ArmHandAgent .....	47
3.3.2. Класс RobotController.....	48
3.3.3. Класс ArticulationJointController .....	48
3.4. Задача перемещения объекта агентом в указанную зону.....	49
3.4.1. Описание смоделированной задачи .....	49
3.4.2. Результаты обучения интеллектуальных агентов.....	50
3.5. Модель манипулятора.....	52
3.6. Результаты обучения по захвату манипулятором объекта .....	54
3.6.1. Описание обучающей среды.....	54
3.6.2. Результаты обучения .....	57
3.7. Результаты обучения по поднятию объекта манипулятором .....	63
3.8. Выводы по разделу .....	64
4. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение	66
4.1. Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения .....	66
4.1.1. Потенциальные потребители результатов исследования .....	66
4.1.2. Технология QuaD .....	67
4.1.3. SWOT-анализ.....	69

4.2. Планирование научно-исследовательских работ .....	70
4.2.1. Структура работ в рамках научного исследования .....	70
4.2.2. Определение трудоемкости выполнения работ .....	71
4.2.3. Разработка графика проведения научного исследования .....	71
4.2.4. Бюджет научно-технического исследования .....	74
4.3. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования ..	77
4.4. Выводы по разделу .....	80
5. РАЗДЕЛ «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ» .....	81
5.1. Введение .....	81
5.2. Правовые и организационные вопросы обеспечения безопасности ....	81
5.2.1. Правовые нормы трудового законодательства .....	81
5.2.2. Организационные мероприятия при компоновке рабочей зоны ...	82
5.3. Профессиональная социальная безопасность .....	84
5.3.1. Анализ вредных и опасных факторов, которые может создать объект исследования. ....	84
5.3.2. Анализ вредных и опасных факторов, которые могут возникнуть на рабочем месте при проведении исследований. ....	84
5.4. Экологическая безопасность .....	92
5.4.1. Анализ влияния объекта исследования на окружающую среду ....	92
5.4.2. Анализ влияния процесса исследования на окружающую среду ..	92
5.5. Безопасность в чрезвычайных ситуациях .....	94
5.5.1. Анализ вероятных ЧС, которые может инициировать объект исследований .....	94
5.5.2. Анализ вероятных ЧС, которые могут возникнуть на рабочем месте при проведении исследований .....	94
5.6. Выводы по разделу .....	96
Заключение .....	97
Список литературы .....	98
Приложение А .....	102
Introduction .....	103
1. The structure of the control system for robotic unmanned mechanisms .....	104
1.1. Unmanned mechanisms control system .....	104
1.2. Training algorithms in a simulated environment .....	105
1.3. Randomizing simulated data .....	106
1.4. Classification and localization of objects .....	107
1.4.1. Localization and classification using depth maps .....	108
1.4.2. Localization and classification without using depth maps .....	110
1.5. Hybrid control system architecture .....	112

## Введение

В современном мире беспилотные роботизированные механизмы выполняют огромный спектр задач, которые могут быть рутинными, биологически или физически опасными, иметь низкую производительность или быть слишком дорогостоящими при использовании традиционных методов. Многие задачи могут выполняться параллельно, поэтому естественным шагом является развертывание нескольких автономных устройств. Группа манипуляторов может эффективно выполнять широкий класс задач, таких как сортировка, транспортировка, ассистирование на производстве, в лабораториях или в быту, работа в опасных местах. Однако для этого требуется высокоточная автономизация аппаратов, а традиционные методы являются малоэффективными в широком диапазоне задач. Для улучшения эффективности, в ряде таких задач можно использовать искусственные нейронные сети [1].

Искусственные нейронные сети (ИНС), получившие в последнее время широкое распространение, являются инструментом, с помощью которого можно эффективно решать очень широкий круг задач. В частности, это распознавание образов и речи, прогнозирование временных рядов, перевод текста, постановка медицинского диагноза, поведенческие алгоритмы в играх и т.д. С каждым годом этот список растёт так-как ИНС, будучи более эффективными, вытесняют традиционные алгоритмы. Робототехника в этом развитии не становится исключением.

Целью данной работы является разработка алгоритма автоматического управления роботизированными механизмами с применением искусственных нейронных сетей.

Для достижения заданной цели были установлены следующие задачи:

- 1) изучить структуру системы управления роботизированными механизмами;



- 2) изучить существующие модели и методы обучения нейронных сетей
- 3) исследовать особенности обучения ИНС для управления роботизированными механизмами;
- 4) разработать программные средства для моделирования работы манипулятора на основе ИНС.

## **1. Структура системы управления автономными роботизированными механизмами**

В данном разделе рассматриваются особенности систем управления автономными аппаратами, различные способы восприятия окружающей среды, методы обучения поведенческим алгоритмам, а также архитектура гибридной системы управления.

### **1.1. Система управления автономным аппаратом**

Система управления автономным аппаратом в общем случае выполняет следующие функции:

- 1) восприятие и распознавание поступающей информации;
- 2) взаимодействие с оператором(человеком);
- 3) планирование и перепланирование поведения;
- 4) управление выполнением действий;
- 5) управление приводами;
- 6) взаимодействие с другими роботами.

Какие из перечисленных функция используются в том или ином аппарате зависит от самого аппарата. Использование ИНС добавляет ещё одну важную функцию – способность и необходимость её обучать.

Для использования нейронных сетей с целью управления автономным аппаратом необходимо решить несколько естественных проблем: минимизация входных данных и сбор обучающей выборки.

Первой проблемой является огромный поток входящих данных. Использование автономных механизмов всегда подразумевает, что аппарат будет взаимодействовать с окружающим миром. Лучшим способом восприятия является визуальный, но обработка визуальной информации с помощью нейронных сетей является крайне ресурсоёмкой задачей.

Второй проблемой является сбор обучающей выборки. Но для сложных или новых систем не существует заранее собранной информации, а её сбор является очень трудоёмким процессом.

## **1.2. Обучение алгоритмов в смоделированной среде**

Сбор обучающих данных для обработки информации с датчиков восприятия окружения или для обучения поведенческих алгоритмов в смоделированной реалистичной среде может значительно увеличить влияние методов машинного обучения на робототехнику, обеспечивая более быстрый, масштабируемый и недорогой способ сбора данных, нежели это возможно в реальной среде.

Особенно многообещающе выглядят методы глубокого обучения с подкреплением, последние результаты которых позволяют добиться уровня производительности, сравнимого с возможностями человека, в таких задачах как Atari [2] и управление роботами [3]. При этом требуется порядка сотни тысяч или миллиона обучающих примеров [2], сбор которых может занять тысячи часов, что делает его непрактичным для многих устройств и задач. При этом глубокое обучение с подкреплением в процессе сбора обучающих примеров выполняет случайные действия, которые могут вывести из строя физическое оборудование. Поэтому возможность изучать линии поведения, полностью смоделированные и разработанные в виртуальных средах, дает преимущество в скорости и безопасности обучения роботизированных механизмов.

К сожалению, расхождения между симуляторами физики и реальным миром затрудняют перенос моделей поведения из симуляции. Процесс настройки параметров моделирования в соответствии с поведением физической системы занимает много времени и подвержен ошибкам. Даже в хорошо проработанной моделированной системе отсутствуют некоторые физические эффекты, такие как гибкость, люфт, износ, которые не улавливаются современными симуляторами физики. Ко всему прочему,

имитирующие датчики с низкой точностью, такие как камеры, зачастую не способны воспроизвести реалистичность и шумы, присутствующие на реальных устройствах. Эти различия, известные под общим названием разрыв реальности (reality gap), создают препятствие для использования смоделированных данных на реальных роботах.

Один из подходов преодоления расхождений между смоделированными данными и реальными состоит в том, чтобы симулятор как можно больше соответствовал физической реальности с использованием высококачественного рендеринга. Использование только реалистичного рендеринга RGB было исследовано в [4] и показало не плохие результаты. Использование реалистичных данных о глубине из 3D-моделей путем всестороннего моделирования таких факторов, как шум сенсоров, коэффициент отражения материалов и геометрию поверхностей позволяло моделям, обученным на визуализированных изображениях, достаточно хорошо переноситься в реальный мир [5].

### **1.3. Рандомизация смоделированных данных**

Другой перспективный подход к преодолению расхождений между смоделированными и реальными данными состоит в том, что вместо сбора данных для обучения в одной моделируемой среде, мы добавляем элемент случайности в симулятор с целью представить модели в широком диапазоне сред во время обучения. Итогом данной процедуры при достаточной изменчивости в симуляции станет модель, которая эффективно извлекает важнейшие признаки как в симуляции, так и на реальном оборудовании без дополнительного обучения.

Цель внесения случайности в элементы симуляции — обеспечить достаточную смоделированную изменчивость во время обучения, чтобы во время тестирования модель могла быть обобщена для реальных данных. Пример данной процедуры представлен на рисунке 1. Слева представлены сцены, смоделированные при различных параметрах, на основе которых

обучаются искусственные нейронные сети. Справа продемонстрирован реальный макет для тестирования обученных сетей [6].

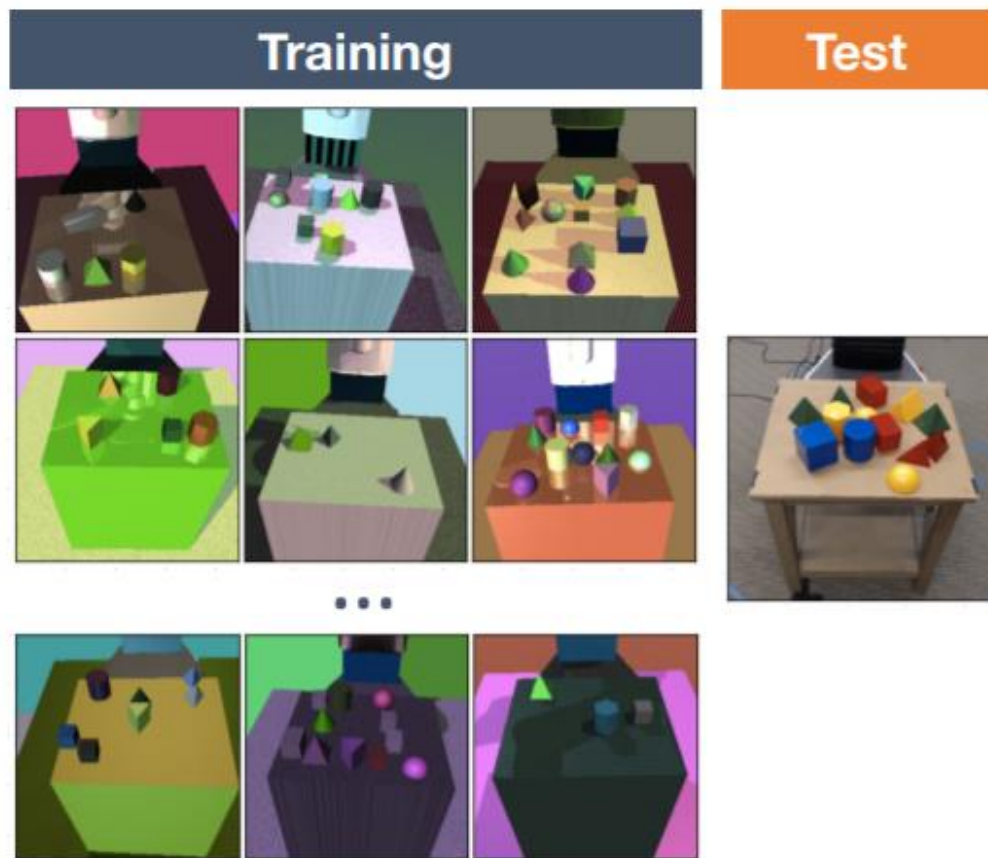


Рисунок 1. Иллюстрация подхода рандомизации симуляции [6]

В процессе рандомизации могут использоваться следующие аспекты предметной области:

- форма и размер объектов;
- количество отвлекающих объектов;
- положение объектов;
- текстуры объектов;
- текстуры земли, стен, фона;
- положение, ориентация и поле зрения камеры;
- количество источников света;
- положение, ориентация и характеристики источников света;
- тип и количество случайного шума, добавленного к датчикам.

Итогом данной процедуры при достаточной изменчивости в симуляции станет модель, обученная в симуляторе, и имеющая достаточно обобщённого опыта для работы в реальном мире без дополнительного обучения.

Рандомизация предметной области – многообещающее направление исследований, направленное на преодоление разрыва между реальностью и разработку алгоритмов поведения роботов, изучаемых в процессе моделирования. Глубокое обучение с подкреплением может позволить изучить более сложные алгоритмы поведения в моделировании посредством крупномасштабного исследования и оптимизации, а рандомизация предметной области может стать важным инструментом для использования таких политик на реальных роботах.

#### **1.4. Локализация и классификация объектов**

Распознавание объектов в неструктурированных сценах – сложная область текущих исследований компьютерного зрения. Одним из важных направлений является робототехника, где способность быстро и точно идентифицировать интересующие объекты имеет решающее значение для универсальных роботов, выполняющих задачи в неструктурированной повседневной среде, такой как дом, офис или склад.

Специфическая проблема восприятия для робототехники имеет ряд уникальных особенностей, которые отличают ее от других проблем компьютерного зрения. Обычная система распознавания объектов должна иметь дело с огромным количеством различных объектов. Один из способов справиться с этим – ввести иерархию, выполняя распознавание на уровне категории, а не на уровне экземпляра. Задача обобщения от нескольких экземпляров до целой категории объектов остается сложной, и существуют многочисленные тесты и задачи, такие как Caltech 256 [7] и PASCAL VOC [8], чтобы способствовать прогрессу в этой области. С другой стороны, для конкретного робота в конкретной среде количество уникальных объектов относительно невелико (возможно, порядка сотен). Это позволяет

рассматривать задачу как проблему распознавания экземпляров, собирая большой объем обучающих данных для каждого объекта. Кроме того, робот может использовать данные, полученные от различных сенсорных устройств, таких как камеры и датчики глубины.

Локализация объектов в робототехнике также представляет дополнительные проблемы, которых нет в тестах распознавания объектов на уровне категорий. Среды реального мира очень загромождены, содержат много перекрытий и часто содержат большее количество различных объектов в одной сцене. Роботам часто приходится взаимодействовать с одними объектами в своей среде, обходя другие. Это означает, что роботизированная система восприятия должна точно локализовать объекты после их обнаружения. Кроме того, чтобы робот мог быстро реагировать на изменения в окружающей среде, роботизированная система восприятия должна работать в режиме реального времени.

#### **1.4.1. Локализация и классификация с применением карт глубины**

В настоящее время существует ряд различных подходов к распознаванию и локализации объектов в сложных неструктурированных сценах. Первым является использование сенсоров для сбора цветных и глубинных изображений. Примерами данных сенсоров являются различные лидары, или более простые датчики по типу Kinect [9], состоящие из инфракрасного проектора, объединенного с монохромной КМОП-матрицей, что позволяет получать трёхмерное изображение при любом естественном освещении (рисунок 2).

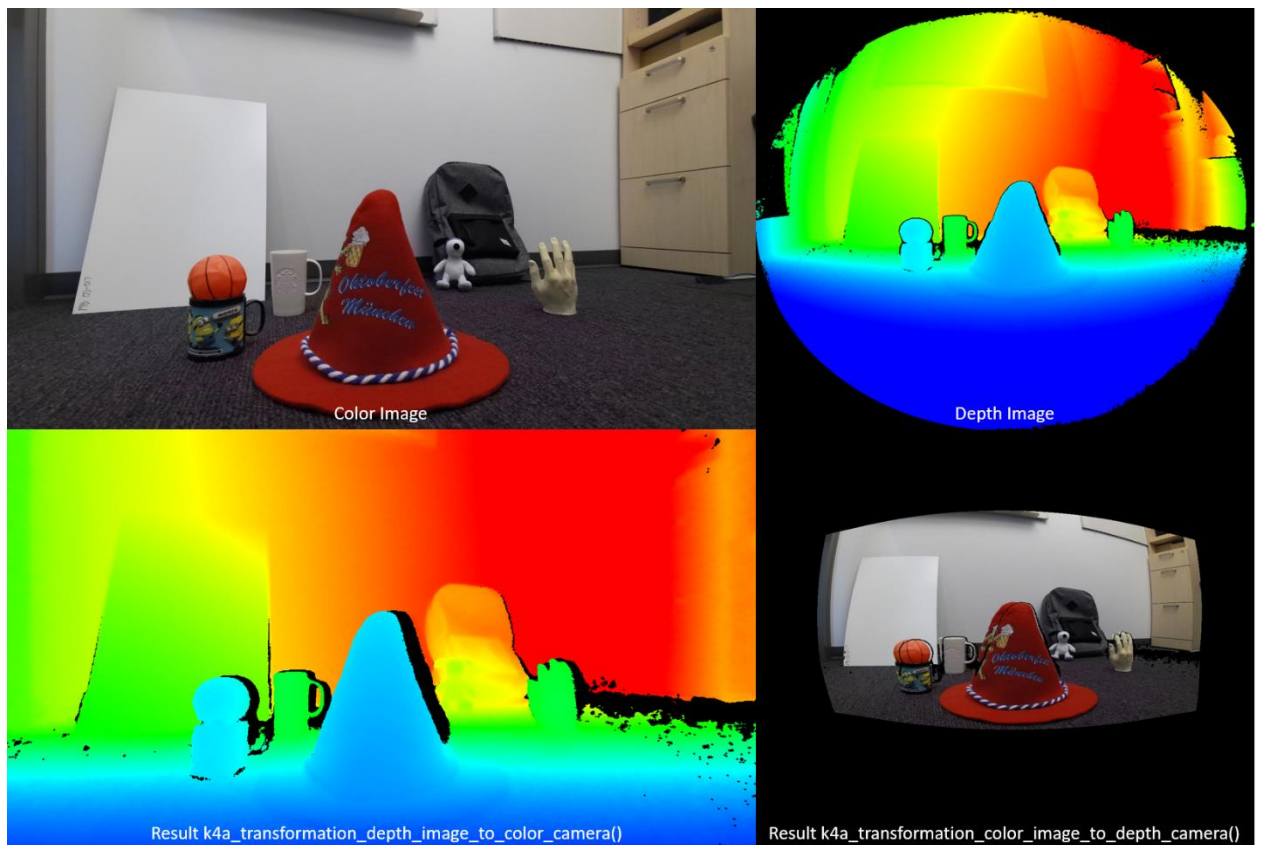


Рисунок 2. Полученные данные с устройства Kinect

Во время обучения распознаванию и локализации объектов на основе информации о глубине получается более точно выделить контуры каждого объекта. На основе контуров, цвета и глубины строится полное облако точек, из которого выделяются трехмерные сеточные модели объектов. Затем, из полученных моделей извлекаются и регистрируются локальные особенности объектов [10].

Во время тестирования, учитывая цветное изображение и облако точек, сцена сегментируется по отдельным обнаруженным объектам. Для каждого сегментированного объекта происходит классификация на основе извлеченных глобальных и локальных признаков, полученных на этапе обучения. На основе информации о глубине подбирается поддерживающая плоскость. После происходят различные проверки согласованности сцены с обнаруженными объектами с целью исключения ошибок классификации, исключаются гипотезы с геометрически несовместимыми объектами, происходит уточнение сегментации объектов. Данный процесс может



повторяться несколько раз, пока сегменты сцены не прекратят перестраиваться и калиброваться [10].

Данный алгоритм занял первое место в конкурсе по распознаванию экземпляров Solutions in Perception Challenge, который проводился Willow Garage на ICRA 2011 [11].

#### **1.4.2. Локализация и классификация без применения карт глубины**

Задача оценки позиционирования модели объекта по одному изображению является хорошо изученной проблемой в литературе. В случае точечных функций она известна как проблема точки перспективы (Perspective-n-Point (PnP)) [12]. Предполагая, что известно достаточно точных соответствий между функциями 2D-изображения и характеристиками 3D-модели, нужно только использовать выбранный алгоритм решения PnP, чтобы получить оценку позиционирования объекта.

Задача распознавания одного объекта и определения его позиции по одному изображению требует решения двух подзадач:

- нахождение соответствий между характеристиками изображения и характеристиками модели;
- оценка позы модели, которая лучше всего согласуется с этим набором соответствий.

Но даже при наличии хорошо различимых инвариантных признаков, таких как SIFT[13] или SURF[14], несовпадающие соответствия неизбежны, что вынуждает использовать более надежные методы оценки в большинстве современных систем распознавания объектов. Исчерпывающий обзор методов распознавания и отслеживания трехмерных объектов на основе моделей доступен в [15].

Проблема распознавания 3D-объектов на основе моделей появилась довольно давно. По данной тематике написана обширная литература, но она

далека от решения в ее наиболее общей форме. Существуют две проблемы, которые сильно влияют на производительность любого алгоритма распознавания.

Первая проблема — это сложность сцены. Данная проблема возникает из-за количества признаков, причем оба крайних значения значительно снижают скорость распознавания. С этим связана проблема повторяющихся объектов: неоднозначность сопоставления, вносимая повторяющимися экземплярами объекта, представляет собой огромную проблему для надежных алгоритмов определения поз объектов, поскольку сопоставленные признаки могут принадлежать разным экземплярам объекта, несмотря на их правильность. Часто используются такие модели, как деревья решений или кластеризация пространства изображений, но ложные срабатывания часто возникают из-за того, что алгоритмы не могут справиться с неожиданной сложностью сцены.

Вторая проблема связана с масштабируемостью и задержкой системы. В системах, работающих онлайн, необходимо достичь компромисса между точностью и задержкой, в зависимости от требований для каждой конкретной задачи. В робототехнике время реакции роботов, работающих в динамической среде, часто ограничивается задержкой их восприятия. Увеличение объема входных данных для обработки, по типу увеличения разрешения камеры, или использования нескольких камер, обычно приводит к серьезным штрафам с точки зрения времени обработки.

В общем, алгоритм для классификации объектов и вычисления точного их расположения относительно камеры делится на несколько этапов. Первым этапом является обучение, на котором создаются разреженные трехмерные модели объектов. Для этого делается набор изображений с объектом в разных позах. Далее происходит извлечение надёжных признаков, например SIFT[13] или SURF[14], которые образуют в разреженную трехмерную модель объекта. Далее происходит этап калибровки: правильное выравнивание и масштаб для

каждой модели вычисляются для соответствия реальным размерам объекта и определяют соответствующую систему координат.

Следует отметить, что данный подход сильно зависит от хорошей калибровки моделей, особенно с точки зрения масштабирования, так как глубина оценивается только основе масштаба каждой модели. Поэтому при создании моделей необходимо проявлять особую осторожность, чтобы установить надлежащий масштаб. К примеру, ошибка моделирования в 1 мм для объекта в 10 см создаёт погрешность определения глубины до 3 см на расстоянии 1 м, что достаточно велико, чтобы вызвать проблемы для роботоманипулятора [16].

Распознавание объектов по локальным признакам на изображениях происходит в несколько этапов. Первым является нахождение соответствий. Данный этап относится к точному сопоставлению характеристик изображения с характеристиками, принадлежащими определенному объекту. Далее идёт процесс оценки позы объекта, которая будет геометрически согласованной с найденными соответствиями. После этого с учётом масштаба объекта в базе данных и на изображении вычисляется точное расстояние до объекта.

Важно отметить, что выбор камеры и объектива может сильно повлиять на точность оценки позы. Тем не менее, камеры становятся более качественными и дешёвыми, а их габариты уменьшаются. Это приводит к тому, что становятся легко доступны параллельное использование нескольких камер для получения вида сцены с разных ракурсов и высоким разрешением. Данный подход эффективен, потому что он обеспечивает оценку глубины, устойчивость к окклюзиям прямой видимости и увеличенное эффективное поле обзора.

Кроме того, более высокое разрешение может потенциально улучшить распознавание сложных объектов и точность алгоритмов оценки позы. но часто с большими штрафными затратами. Дополнительное разрешение часто приводит к увеличению количества ложных срабатываний, а также к

серьезному ухудшению условий, задержки и снижению пропускной способности.

Помимо всего вышесказанного, использование нескольких камер позволяет проводить реконструкцию запечатленных сцен в виртуальной среде (рисунок 3). Это может значительно упростить процесс обучения интеллектуальных агентов по взаимодействию в окружающими объектами.



Рисунок 3. (Колонки 1-3) Локализация и классификация объектов по изображению, (Колонка 4) показывает реконструкцию данных сцен в виртуальной среде.

### 1.5. Архитектура гибридной системы управления

С целью уменьшения объёма входных данных на систему, отвечающую за выполнения каких-либо действий, можно использовать гибридный подход к реализации систем искусственного интеллекта [17]. Данный метод основан на разделении общей системы управления на логическую и ассоциативную части (рисунок 4).

В данном представлении ассоциативный уровень обрабатывает информацию, полученную от внешней среды. Под информацией подразумевается любые данные: изображение, датчики приближения, уровень кислорода, температура, тепловые датчики, информация, полученная от других аппаратов, команды от оператора и т.д. Вся полученная информация обрабатывается для созданий ассоциаций различных объектов и дальнейшей их классификации, кластеризации и обобщения. Набор информации зависит

от целей, которые должен выполнять аппарат. Обработка информации может полностью или частично заниматься ИНС.

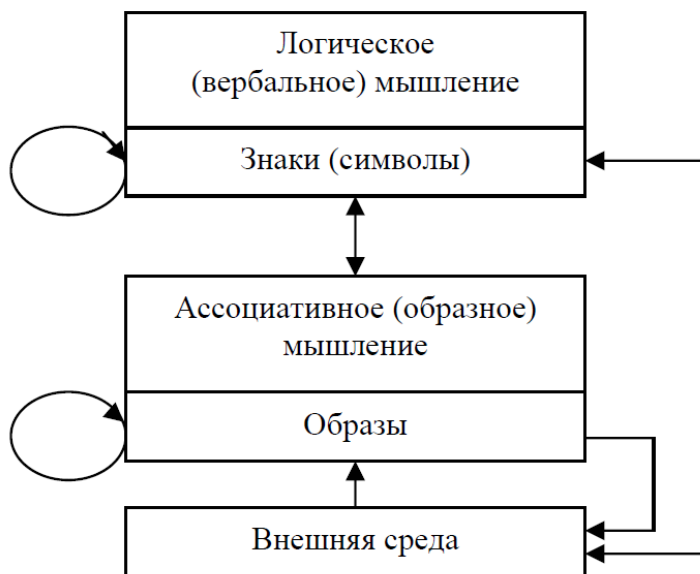


Рисунок 4. Два уровня интеллектуальной системы

Логический уровень обрабатывает информацию, полученную от ассоциативного уровня, и на её основе принимает какие-либо действия во внешней среде. На этом уровне также могут использоваться как классические методы инженерии, так и искусственные нейронные сети.

Данный подход позволяет разделить задачу по управлению автономным аппаратом на группу подзадач. Это особенно актуально при использовании ИНС так как обучить группу небольших нейронных сетей, каждая из которых будет выполнять свою подзадачу намного проще в плане сбора обучающей выборки.

Самый распространённый пример использования гибридной системы управления – это применения лидаров и камер для детекции объектов и построения виртуальной среды на ассоциативном уровне обработки информации, и любые другие алгоритмы на логическом (рисунок 5). Данный подход способствует не только разделению задачи управления автомобилем, но и позволяет тестировать, а если речь идёт об ИНС, то и обучать поведенческие алгоритмы на моделях. Модели в свою очередь должны лишь

имитировать информацию, генерирующую ассоциативным уровнем, а не воссоздавать реалистичную окружающую среду.

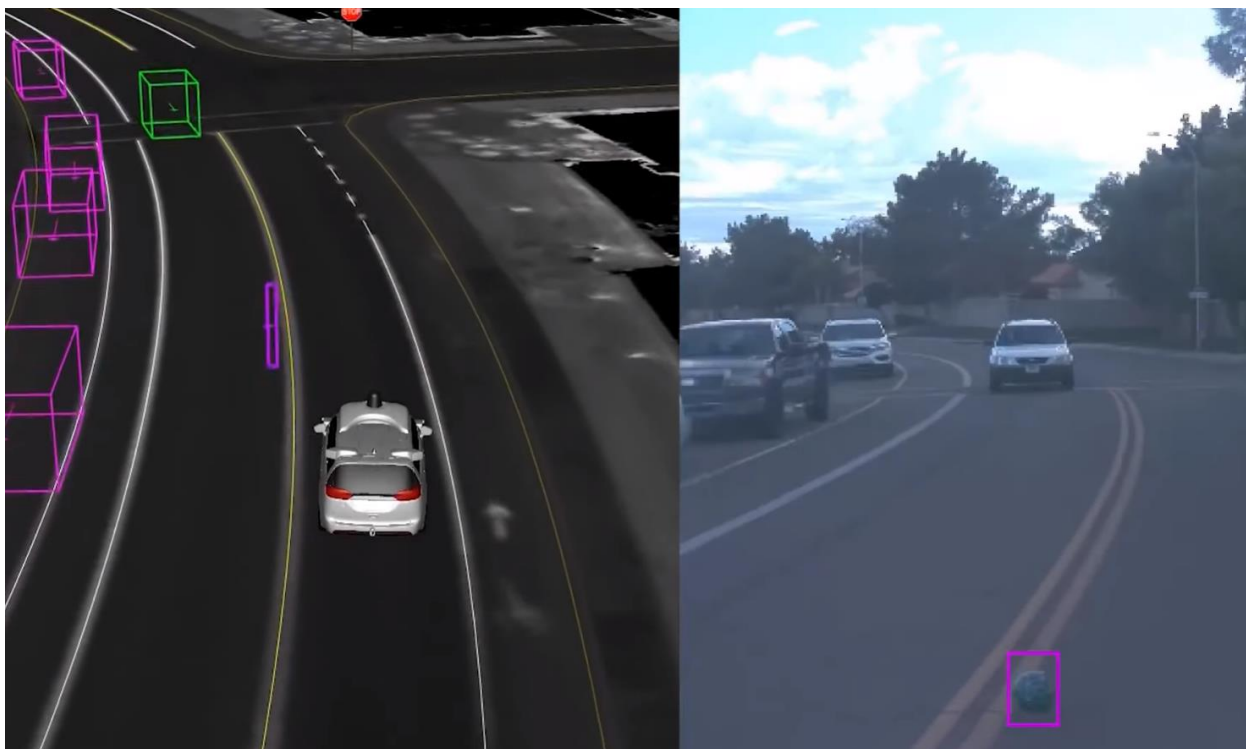


Рисунок 5. Пример ассоциативного преобразования информации

В данной работе будет рассматриваться возможности использования ИНС для обработки информации на логическом уровне. Информация для обучения будет собираться при помощи модели, имитирующей похожие процессы.

## **2. Общие принципы работы искусственных нейронных сетей**

В данном разделе описываются общие принципы работы искусственных нейронов и нейронных сетей.

### **2.1. Общая информация**

Искусственная нейронная сеть (ИНС) – математическая модель, а также её программное или аппаратное воплощение, построенная по принципу организации и функционирования биологических нейронных сетей — сетей

нервных клеток живого организма. Элементарной единицей ИНС является искусственный нейрон [18].

Входной информацией для искусственного нейрона (рисунок 6) является произведения признаков ( $X_i$ ) на соответствующие им веса ( $W_i$ ). Полученные результаты складываются сумматором, и передаются в качестве аргумента на функцию активации  $f$ . Значение этой функции и есть выход нейрона ( $Y$ ).

В модель нейрона, показанную на рисунке 6, включен пороговый элемент, который имеет символическое обозначение  $b_k$ , и отражает увеличение или уменьшение входного сигнала, подаваемого на функцию активации [18].

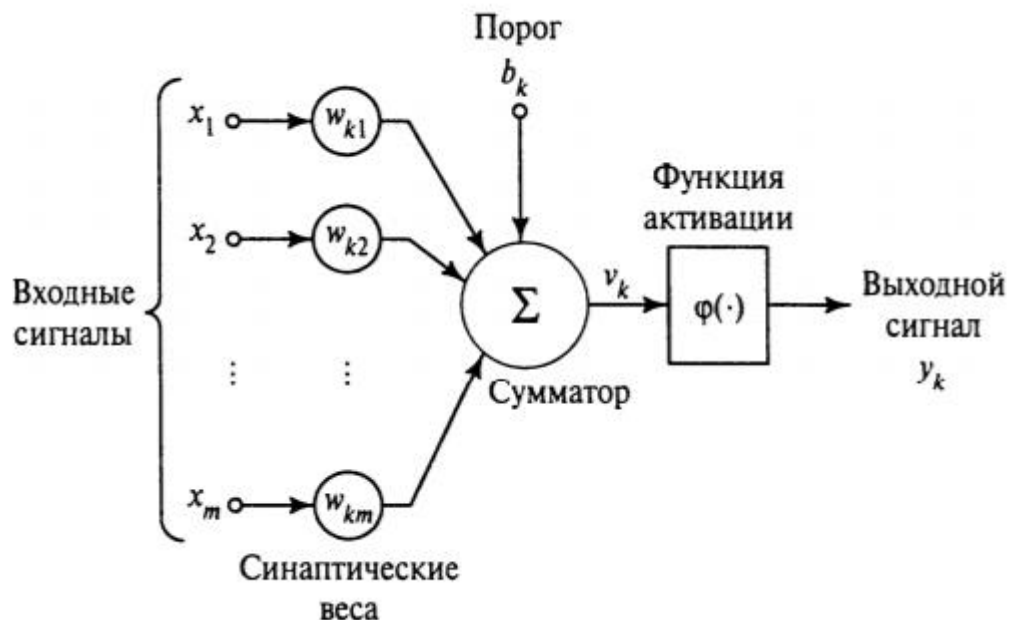


Рисунок 6. Нелинейная модель нейрона

Существует огромное количество архитектур нейронных сетей: однослойные и многослойные сети прямого распространения, рекуррентные, рекурсивные, сети долгой краткосрочной памяти и т.д. В данной работе будут рассматриваться сети прямого распространения.

ИНС прямого распространения характеризуется тем, что сигнал распространяется строго от входного слоя к выходному. Данная архитектура состоит из 3 типов слоёв: входных (input), скрытых (hidden) и выходных (output) (рисунок 7). Корректная работа ИНС подразумевает, что что при

получении каких-либо признаков на вход, на выходе будет получен результат. Соответственно ИНС должна иметь два и больше слоёв. Как правило, количество скрытых слоёв зависит от сложности задачи, и каждый скрытый слой производит переход из одного пространства признаков в другой.

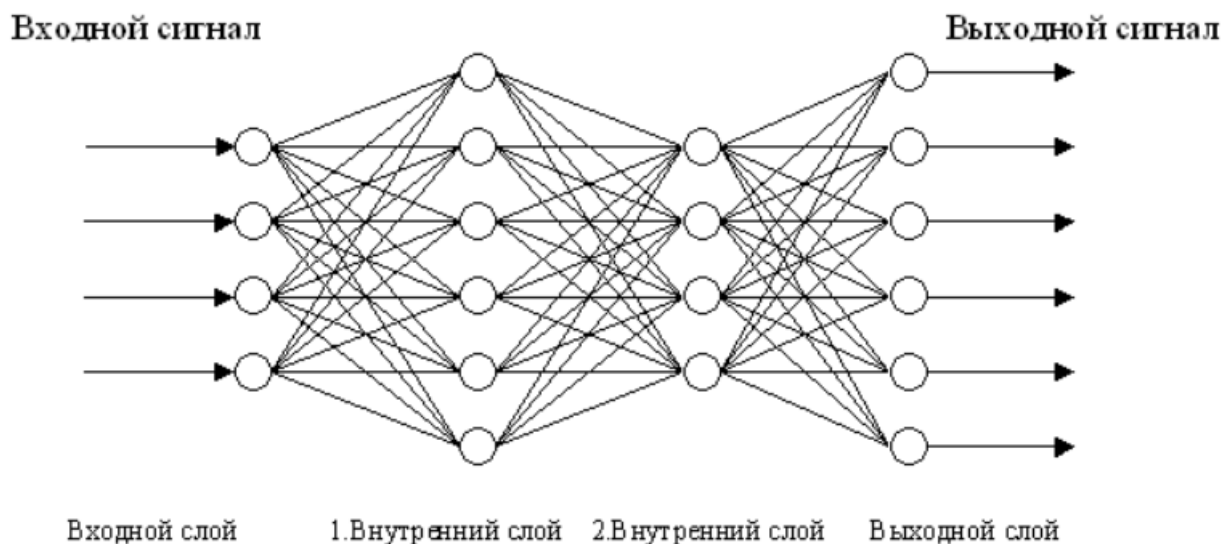


Рисунок 7. Базовая структура ИНС прямого распространения

## 2.2. Принцип работы искусственных нейронных сетей

Как было сказано выше, ИНС состоит из трёх типов слоёв. ИНС может иметь различное количество входов, выходов и скрытых слоёв. Архитектура сети зависит от решаемой задачи.

При обработке информации ИНС, данные подаются на входной слой, и с него идут на первый скрытый слой и при этом умножаются на вес связи. На первом скрытом слое каждый нейрон суммирует пришедшие на него произведения признаков и весов. Затем данная сумма подаётся как аргумент в функцию активации, и значение функции и будет являться выходным значением нейрона.

Существует множество различных функций активации, имеющих различные области значений, и лучше предрасположенных к разного рода задачам. Приведу несколько примеров: на рисунке 8 изображена сигмоида. Её областью выходных значений является число от 0 до 1.



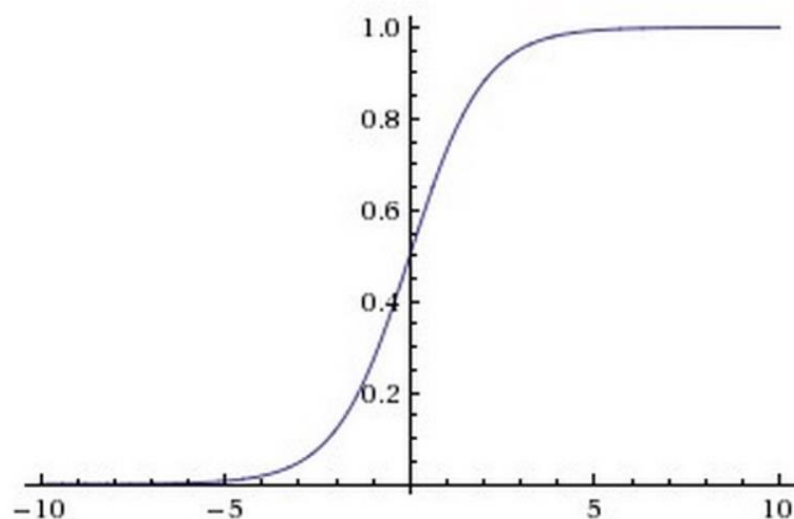


Рисунок 8. График значений сигмоиды

На рисунке 9 изображен гиперболический тангенс с областью выходных значений от -1 до 1.

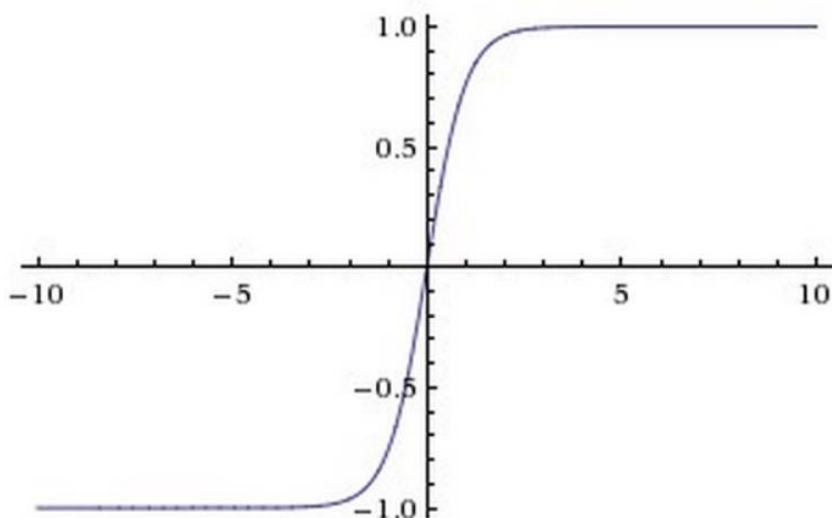


Рисунок 9. График значений гиперболического тангенса

Выдача нейроном выходного значения называется активацией нейрона. После того, как все нейроны первого скрытого слоя будут активированы, они так же передадут признаки, но уже новые, сформированные ими, следующему слою, и так вплоть до выходного.

Для корректной работы ИНС, её необходимо обучать. Обучение бывает с учителем, без учителя и с подкреплением.

### 2.3. Парадигма обучения с учителем

Обучение с учителем (Supervised learning) – один из разделов машинного обучения, посвященный решению задачи нахождения зависимостей между входным и выходным векторами данных. Алгоритм подразумевает, что при обучении известны и входные, и выходные вектора. Вместе они называются представительской или обучающей выборкой, и на их основе сеть будет обучаться, обобщая получаемую информацию в процессе корректировки синоптических весов [18].

Блочная диаграмма, иллюстрирующая данную форму обучения представлена на рисунке 10. Понятие учителя можно рассматривать как наличие знаний об окружающей среде. При этом сама среда неизвестна обучаемой нейронной сети. В представленной диаграмме, обучаемой сети и учителю поступает информация из окружающей среды. На основе встроенных знаний учитель может сформировать и передать обучаемой нейронной сети желаемый отклик, соответствующий данному входному вектору. Этот желаемый результат представляет собой оптимальные действия, которые должна выполнить нейронная сеть. Параметры сети корректируются с учетом обучающего вектора и сигнала ошибки. Сигнал ошибки (error signal) — это разность между желаемым сигналом и текущим откликом нейронной сети. Корректировка параметров выполняется пошагово с целью имитации (emulation) нейронной сетью поведения учителя. Эта эмуляция в некотором статистическом смысле должна быть оптимальной. Таким образом, в процессе обучения знания учителя передаются в сеть в максимально полном объеме. После окончания обучения учителя можно отключить и позволить нейронной сети работать со средой самостоятельно [18].

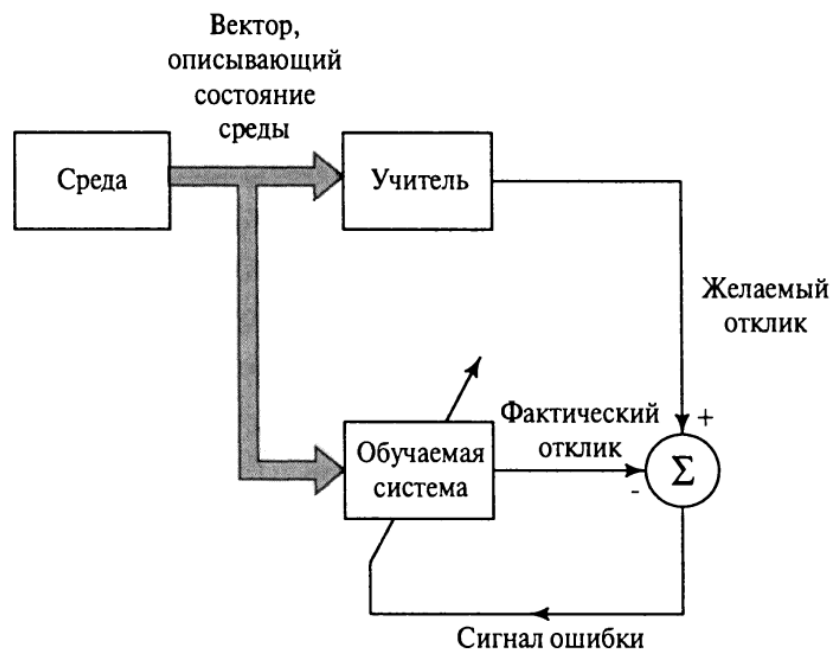


Рисунок 10. Блочная диаграмма обучения с учителем

Описанная форма обучения с учителем является ничем иным, как обучением на основе коррекции ошибок. Это замкнутая система с обратной связью, которая не включает в себя окружающую среду. Производительность такой системы можно оценивать в терминах среднеквадратической ошибки или суммы квадратов ошибок на обучающей выборке, представленной в виде функции от свободных параметров системы. Для такой функции можно построить многомерную поверхность ошибки (error surface) в координатах свободных параметров. При этом реальная поверхность ошибки усредняется (averaged) по всем возможным примерам, представленным в виде пар «вход-выход». Любое конкретное действие системы с учителем представляется одной точкой на поверхности ошибок. Для повышения производительности системы во времени значение ошибки должно смещаться в сторону минимума на поверхности ошибок. Этот минимум может быть, как локальным, так и глобальным. Это можно сделать, если система обладает полезной информацией о градиенте поверхности ошибок, соответствующем текущему поведению системы. Градиент поверхности ошибок в любой точке — это вектор, определяющий направление наискорейшего спуска по этой поверхности. В случае обучения с учителем на примерах вычисляется моментальная оценка (instantaneous estimate) вектора градиента, в которой

входной вектор считается функцией времени. При использовании результатов такой оценки перемещение точки по поверхности ошибок обычно имеет вид «случайного блуждания». Тем не менее при использовании соответствующего алгоритма минимизации функции стоимости, адекватном наборе обучающих примеров в форме «вход-выход» и достаточном времени для обучения системы обучения с учителем способны решать такие задачи, как классификация образов и аппроксимация функций [18].

## **2.4. Парадигма обучения без учителя**

Описанный выше процесс обучения происходит под управлением учителя. Альтернативная парадигма обучения без учителя (learning without a teacher) самим названием подчеркивает отсутствие руководителя, контролирующего процесс настройки весовых коэффициентов. При использовании такого подхода не существует маркированных примеров, по которым проводится обучение сети [18].

Обучение без учителя (unsupervised) (или обучение на основе самоорганизации (self-organized)) осуществляется без вмешательства внешнего учителя, или корректора, контролирующего процесс обучения (рисунок 11). Существует лишь независимая от задачи мера качества (task-independent measure) представления, которому должна научиться нейронная сеть, и свободные параметры сети оптимизируются по отношению к этой мере. После обучения сети на статистические закономерности входного сигнала она способна формировать внутреннее представление кодируемых признаков входных данных и, таким образом, автоматически создавать новые классы [18].



Рисунок 11. Блочная диаграмма обучения без учителя

Парадигма обучения без учителя успешно применяется для кластеризации данных, поиска правил ассоциации, сокращения размерности данных и визуализации данных.

## **2.5. Парадигма обучения с подкреплением**

В данном подразделе описываются общие принципы, подходы и особенности парадигмы «обучение с подкреплением».

### **2.5.1. Общая информация**

С развитием современных технологий человечество ставит себе на службу системы с все более и более возрастающей сложностью. При разработке таких систем возникает проблема оптимального их контроля, которую зачастую невозможно решить реализацией какого-то определенного алгоритма. Для решения подобных задач на стыке областей машинного обучения и оптимального управления была создана область обучения с подкреплением, фокусирующаяся на обучении поведения агента в определенных окружениях для достижения определенного результата. Задачи, в которых эти методы нашли применения, варьируются от задач управления роботами и контроля промышленных процессов до управления портфелями ценных бумаг и участием в компьютерных играх. В последнее время популярными стали алгоритмы обучения с подкреплением, использующие нейронные сети над состояниями окружений.

Задача обучения с подкреплением предполагает наличие агента, получающего определенную информацию о состоянии окружения и

способного совершать действия, оказывающие влияние на окружение. Цель обучения с подкреплением состоит в обучении агента способу получения максимального значения числовой награды, которая выдается агенту при совершении определенных действий.

В работе агентов выделяют важную проблему выбора между эксплуатацией текущих знаний агента для получения наибольшей награды и исследованием окружения с целью получения знаний о структуре окружения. Эти два варианта выбора действий подразумевают под собой одну и ту же итоговую цель, один вариант максимизирует полученную награду в ближайшем будущем, а другой – в долгосрочном периоде.

Пространства состояний окружения можно разделить на две большие группы: дискретные пространства состояний, в которых каждая переменная окружения является конечной дискретной величиной, и непрерывные пространства состояний, в которых хотя бы одна переменная окружения является непрерывной величиной. Мощность множества всех возможных состояний конечна для дискретных пространств состояний и бесконечна для непрерывных пространств состояний.

Существуют стратегии исследования окружений, гарантированно приводящие агента к оптимальной стратегии действий на пространствах с небольшой мощностью множества всех возможных состояний. На задачах с большой или бесконечной мощностью множества всех возможных состояний такие методы становятся неприменимы из-за больших вычислительных затрат. В современных системах обучения с подкреплением применяются простые стратегии исследования окружений, которые не всегда позволяют найти оптимальную стратегию или находят ее слишком долго. Поэтому возникает необходимость в разработке более эффективных алгоритмов исследования окружений.

Одна из трудностей в изучении стратегий исследования окружений состоит в том, что не существует принятого определения для описания

оптимального действия с точки зрения исследования оптимальной структуры окружения. Оптимальное с эксплуатационной точки зрения действие определяется как действие, максимизирующее ожидаемую от этого действия награду. Дополнительные трудности в формализацию понятия исследования окружений вносит тот факт, что многие современные системы обучения с подкреплением не хранят информацию о влиянии действий агента на окружение.

### **2.5.2. Фундаментальные методы решения**

В данном подразделе рассматриваются различные подходы к обучению искусственных нейронных сетей методом «Обучение с подкреплением».

#### **Методы Монте-Карло**

Методы Монте-Карло не предполагают полного знания об окружающей среде, но требуют наличие опытных данных, под которым понимается набор последовательностей состояний, действий и наград, полученных в результате онлайн или смоделированного взаимодействия с окружающей средой. Хотя для смоделированных опытных данных и требуется модель, данной модели требуется только генерировать выборки перемещений, а не полное распределение всех возможных перемещений, как того требует динамическое программирование [19].

Методы Монте-Карло решают задачу обучения с подкреплением, основываясь на средней ценности выборки. Чтобы удостовериться, что ценность будет определенной, методы Монте-Карло определяются только для эпизодических задач. Это предполагает, что опытные данные разделены на эпизоды, и что все эпизоды в итоге заканчиваются, вне зависимости от того, какие действия выбраны. Только после завершения эпизода происходит оценивание значений, и стратегии изменяются.

Важный факт о методах Монте-Карло состоит в том, что оценки для каждого состояния являются независимыми, в отличие от методов динамического программирования. В особенности, следует заметить, что

вычисление оценки отдельного состояния не зависит от числа состояний. Это делает методы Монте-Карло особенно привлекательными в тех случаях, когда требуется оценить только некоторое подмножество состояний.

Если модель недоступна, то тогда полезно получить оценки действий, а не состояний. При наличии модели вполне достаточно иметь оценки состояний, чтобы определить стратегию. Для этого необходимо просмотреть на один шаг вперед и выбрать, какое действие приводит к наилучшей комбинации вознаграждения и следующего состояния. При отсутствии модели, однако, одних оценок состояния недостаточно. Необходимо в явном виде оценить значение каждого действия для того, чтобы в дальнейшем полученные оценки использовать для построения стратегии.

Задача оценки стратегии состоит в вычислении функции  $Q_{\pi}(s, a)$  – ожидаемой награды при начале в состоянии  $s$ , выполнении действия  $a$ , и следуя стратегии  $\pi$ . Данная задача является основной задачей обучения с подкреплением.

Общая идея формирования управления состоит в действиях по схеме, которая опирается на понятие обобщенной итерации по стратегиям (ОИС). В концепции ОИС используется как приближенная стратегия, так и приближенная функция ценности. Коррекция функции ценности происходит таким образом, чтобы наилучшим образом отвечать текущей стратегии, а стратегия постоянно уточняется в зависимости от текущей функции ценности.

Как правило в методах Монте-Карло чередуются операции оценивания и улучшения от эпизода к эпизоду. После завершения эпизода происходит оценивание стратегии, используя награду, которая была получена в эпизоде, после чего стратегия улучшается. Алгоритм, соответствующий данной схеме, получил название ИС-алгоритм Монте-Карло с изучающими стартами.

Обобщая вышесказанное, следует сказать, что методы Монте-Карло имеют 3 вида преимуществ по сравнению с методами динамического программирования. Во-первых, оптимальное поведение можно построить в



результате непосредственного взаимодействия с окружающей средой, при этом не требуется знания модели динамики окружающей среды. Во-вторых, данные методы можно использовать в сочетании с методами моделирования, позволяющими получить смоделированные совокупности эпизодов. В-третьих, методы Монте-Карло могут легко и эффективно фокусироваться на небольшом подмножестве состояний.

### **Методы временных различий, Q-learning**

Методы временных различий (TD-методы – Temporal Difference) совмещают в себе идеи методов Монте-Карло и динамического программирования. Как и методы Монте-Карло, TD-методы могут обучаться без модели динамики окружения, а не посредственно из опыта. Как и методы динамического программирования, TD-методы обновляют расчетные оценки, частично основываясь на других оценках, при этом не ожидая финального результата [20].

И TD-методы, и методы Монте-Карло используют опыт, чтобы решить задачу предсказания. Из некоторого данного опыта следования стратегии  $\pi$ , оба метода обновляют их оценки  $V$  функции  $V\pi$  для состояний  $S_t$ , которые не являются финальными и присутствуют в данном опыте. В случае посещения состояния  $S_t$  в момент времени  $t$  оба метода корректируют свои оценки основываясь на том, что случилось после посещения данного состояния.

#### **2.5.3. Функция ценности**

Большинство алгоритмов обучения с подкреплением включает в себя понятие функции ценности. Функция ценности — это функция от состояния (или от пары состояния и действия), которая оценивает, насколько ценно для агента пребывать в заданном состоянии (или насколько ценно применить данное действие в данном состоянии). Понятие о том, насколько ценно какое-то действие или состояние выражается в понятии будущей ожидаемой награды. Награда, которую агент может рассчитывать получить в будущем,

зависит от принимаемых в дальнейшем действий, поэтому функции ценности определены для каждой стратегии принятия решений.

Механизм оценки действий является сложной и неоднозначной процедурой так как вознаграждение может характеризовать действия, выполненные задолго до получения вознаграждения, или верную последовательность действий, которая насчитывает значительное количество шагов.

Существуют различные подходы к данной проблеме. Первый из них – это выдавать вознаграждение плавно и часто. Но при этом перед программистом возникает задача планирования действий, за которые агент будет получать вознаграждения, а также их рационального количества.

Другой перспективный способ улучшения обучения с сильно отсроченным вознаграждением – иерархическое обучение с подкреплением. Иерархическое обучение с подкреплением пытается разложить долгосрочную задачу на ряд целей и подзадач. Разложив проблему, мы эффективно расширяем временные рамки, в которых принимаются решения.

### **3. Разработка программных средств для моделирования работы манипулятора**

#### **3.1 Выбор средств реализации**

Для разработки программных модулей по обучению интеллектуальных агентов управлению роботизированными устройствами, необходимо смоделировать устройство в среде с реалистичной физикой, и разработать конфигурации интеллектуальных агентов. Агенты должны иметь возможность взаимодействовать со средой, получать обратную информацию, и в процессе обучаться. Исходя из вышесказанного, выбор средств реализации начинается с выбора среды разработки, которая имеет возможность моделировать реалистичные физические взаимодействия объектов.

Проанализировав такие среды разработки, как Unity, Unreal Engine, CryEngine, V-REP, а также возможность написания собственного подхода к симуляции физики, был выбран Unity. Основными причинами стали:

- добавление в 2020 году физической библиотеки PhysX 4.0, которая обеспечивает более точное и производительное моделирование реалистичной физики взаимодействия объектов;
- предусмотренная возможность моделирования движений с помощью суставов;
- большое количество библиотек и плагинов, в том числе и по ИНС;
- обширная база документации;
- удобное отслеживание ошибок и работы приложения.

Для обучения нейронной сети был выбран метод обучения с подкреплением. Данный метод показал хорошие результаты в широком круге задач, таких как победа чемпиона мира в Go и обучении играм Atari, что можно считать частными случаями обучения агента взаимодействовать со средой.

После анализа готовых решений выбор сводится к двум языкам программирования: C# и Python.

Python – высокоуровневый язык программирования общего назначения. Среди преимуществ данного языка программирования можно выделить высокую производительность, хорошую читаемость кода, кроссплатформенность, возможность объектно-ориентированного подхода, автоматическое управление памятью, поддержку многопоточковых вычислений, а также большое количество решений, библиотек и документации для работы с нейронными сетями.

C# – объектно-ориентированный язык программирования, созданный как язык разработки приложений для платформы Microsoft .NET Framework. Среди преимуществ данного языка программирования можно выделить хорошую читаемость кода, автоматическое управление памятью, поддержку многопоточковых вычислений, а также большое количество библиотек для решения огромного спектра задач.

Сравнивая языки программирования, Python является фаворитом для обучения нейронных сетей. Среди основных преимуществ можно выделить высокую производительность, большое количество примеров, библиотек, документации.

Так как основная часть работы – это моделирование среды, и обучение агента взаимодействуя со средой выполнять какие-либо запланированные действия, то появляется необходимость отслеживать данные действия визуально. Отсюда следует, что основная работа будет происходить на языке программирования C#, на движке Unity. Но данный язык не поддерживает возможности обучения высокопроизводительных искусственных нейронных сетей, поэтому процесс обучения будет происходить на языке программирования Python. Связь между блоками на разных языках программирования будет основана с помощью набора инструментов для агентов машинного обучения «ML-Agents».

ML-Agents — это проект с открытым исходным кодом, который позволяет играм и симуляторам служить средой для обучения интеллектуальных агентов. Данный проект предоставляет возможность подключения современных алгоритмов на основе библиотеки машинного обучения «PyTorch», и обучения интеллектуальных агентов для игр 2D, 3D и VR / AR. Также предоставляется возможность использования Python API для обучения агентов с использованием обучения с подкреплением, имитационного обучения, нейроэволюции или любых других методов.

### **3.2 Определение требований к проектируемой системе**

Для достижения поставленной цели работы, необходимо провести проектирование программного комплекса, демонстрирующего процесс обучения искусственной нейронной сети заданным поведенческим алгоритмам. В качестве поведенческих алгоритмов были выбраны задачи по перемещению агентом объектов посредством толкания, и по захвату объектов манипулятором.

Исходя из вышесказанного можно сделать вывод, что обучение будет основываться на информации, получаемой агентом от окружающей его среды, и реакции среды на действия агента. Рассмотрим диаграмму, иллюстрирующую данное взаимодействие (рисунок 12). При получении сигнала, запроса на новое состояние, агент получает информацию об окружающей его среде. На основе этой информации он предпринимает какие-либо действия. Среда переходит в новое состояние, и опираясь на функцию ценности вознаграждает агента. Затем происходит визуализация, и модель переходит в состояние ожидания запроса на переход в следующую эпоху.

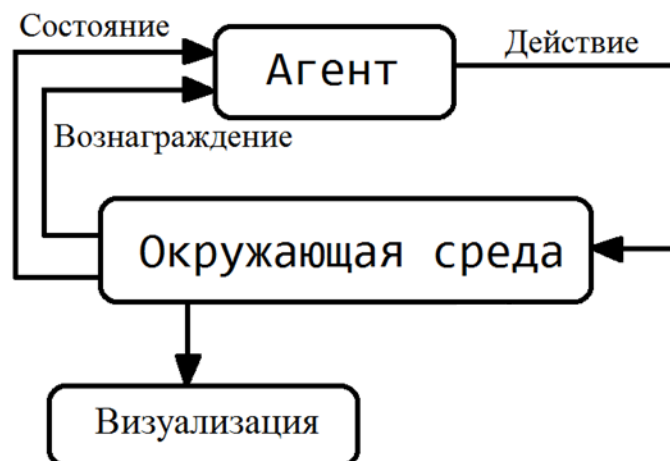


Рисунок 12. Диаграмма взаимодействия элементов системы обучения

Диаграмма на рисунке 12 показывает общие принципы обучения интеллектуальных агентов. Взаимодействие элементов на уровне языков программирования представлена на рисунке 13.

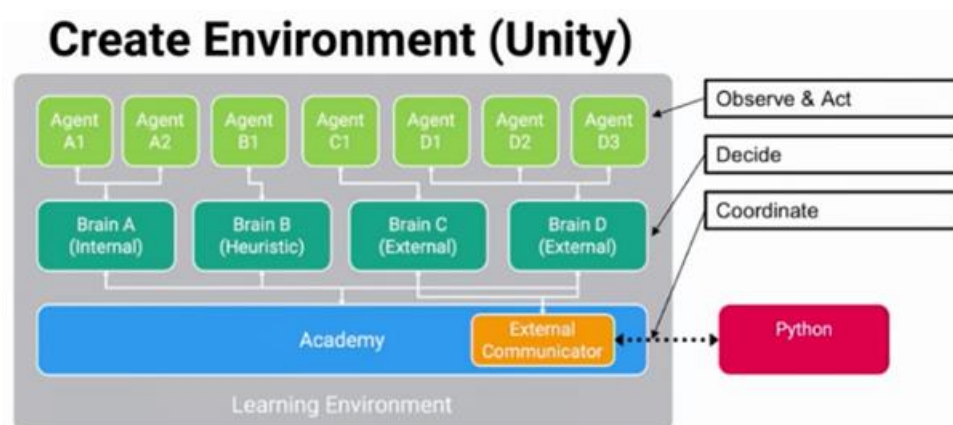


Рисунок 13. Диаграмма взаимодействий элементов системы

Все эксперименты проводились на ноутбуке с характеристиками:

- Процессор Intel® Core™ i5-9300H
- Графический процессор: NVIDIA GeForce GTX 1650
- ОЗУ 8 Гб (1 поток)

### 3.3 Структура приложения

В данном разделе приведены основные методы и классы, отвечающие за работоспособность разработанных программных комплексов.

### 3.3.1. Класс ArmHandAgent

Основная задача класса **ArmHandAgent**: взаимосвязь смоделированной сцены и обучаемого агента, запуск и перезапуск сцены, получение дополнительной информации об окружающей агента среде, а также задание функции вознаграждения за действия. Ниже описаны основные поля и методы:

**refabArena** – содержит объект ограничивающей арены вокруг агента. Используется для случайного расположения объектов при каждой симуляции;

**goal** – объект, характеризующий область, куда агент должен доставить объект;

**block** – объект, который необходимо транспортировать к **goal**;

**armRobot** – объект, управляемый агентом, содержащий набор суставов, соединённых сочленениями;

**armHand** – объект, расположенный на конце последнего сочленения **armRobot**, и позволяющий захватывать объекты;

**target** – не физический объект, расположенный между захватами **armHand**. Используется при расчёте расстояния между захватом и объектом;

**captureDetect** – переменная, характеризующая положение захвата;

**keycodes** – список клавиш для ручного управления;

**ScoredAGoal** – метод, отвечающий за вознаграждение и перезапуск сцены после выполнения поставленного задания;

**MoveAgent** – метод, отвечающий за управление сочленениями. Может получать команды как от пользователя, так и с ИНС;

**OnActionReceived** – метод, отвечающий за функцию вознаграждения агента;

**CollectObservations** – метод, отвечающий за информацию, поступающую от смоделированной среды к агенту;

**Heuristic** – метод, отвечающий за ручное управление агентом;

**ResetBlock** – метод, отвечающий за сброс положения и скорости объекта;

**ResetRobot** – метод, отвечающий за сброс положений сочленений манипулятора.

### 3.3.2. Класс RobotController

Основная задача класса RobotController: хранить все объекты, соединенные сочленениями в манипуляторе. Является связующим элементом между методами, отвечающими за управление, и отдельными суставами.

Ниже описаны основные поля и методы класса RobotController:

**joints** – список всех имеющихся суставов в модели манипулятора;

**GetCurrentJointRotations** – возвращает положение всех имеющихся в манипуляторе сочленений;

**StopAllJointRotations** – метод, отвечающий за остановку движения всех суставов;

**RotateJoint** – метод, отвечающий за движение заданного сустава в заданном направлении;

**ForceJointsToRotations** – метод, отвечающий за параллельное изменение положений нескольких сочленений;

### 3.3.3. Класс ArticulationJointController

Основная задача класса ArticulationJointController: управление отдельными суставами манипулятора.

Ниже описаны основные поля и методы класса:

**RotationDirection** – перечисляемый тип, определяющий необходимость приложения силы к соединению суставов в том, или ином направлении;



**FixedUpdate** – метод, отвечающий за приложение силы к соединению суставов на протяжении некоторого времени;

**CurrentPrimaryAxisRotation** – метод, возвращающий текущее состояние сочленения между выбранными суставами;

**RotateTo** – метод, рассчитывающий новое состояние для соединения суставов с учётом прикладываемых сил.

**ForceToRotation** – метод, позволяющий сменить положение сустава без применения сил. Необходим для возвращения к начальному состоянию;

### **3.4 Задача перемещения объекта агентом в указанную зону**

С целью проверки различных алгоритмов обучения интеллектуальных агентов, с разными вариациями входных параметров, была смоделирована задача перемещения объекта агентом в указанную зону.

#### **3.4.1. Описание смоделированной задачи**

На рисунке 14 продемонстрирована сцена, в которой агентом является синий квадрат. Белым квадратом является объект, который необходимо переместить в зелёную зону. Перемещение объекта возможно только методом его толкания агентом, никаких функций сцепки элементов не предусмотрено. Лучи, исходящие от агента, являются сенсорами, способными определять тип объекта, а также расстояние до него.

Каждая симуляция ограничена по числу пройденных эпох. Это сделано с целью выхода симуляции из тупиковой ситуации, когда объект расположен в углу сцены, и агент не имеет возможности сдвинуть его. В начале каждой новой симуляции агент и объект располагаются в случайных местах.

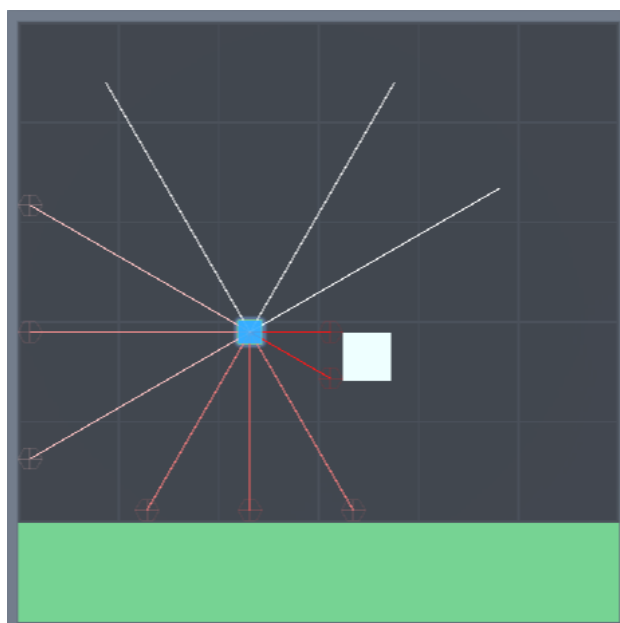


Рисунок 14. Смоделированная среда для обучения интеллектуальных агентов

### 3.4.2. Результаты обучения интеллектуальных агентов

Для достижения разумной степени автономии интеллектуального агента два основных требования — это восприятие и рассуждение. Первый обеспечивается сенсорной системой, которая собирает информацию об агенте относительно окружающей среды. Количество и качество сенсоров может сильно отличаться, и подбирается индивидуально для каждой задачи. На рисунке 15 продемонстрированы различные конфигурации сенсоров, которые отличаются углом обзора и максимальным измеряемым расстоянием. В качестве алгоритмов обучения интеллектуальных агентов применялись Soft Actor-Critic (SAC) и Proximal Policy Optimization (PPO).

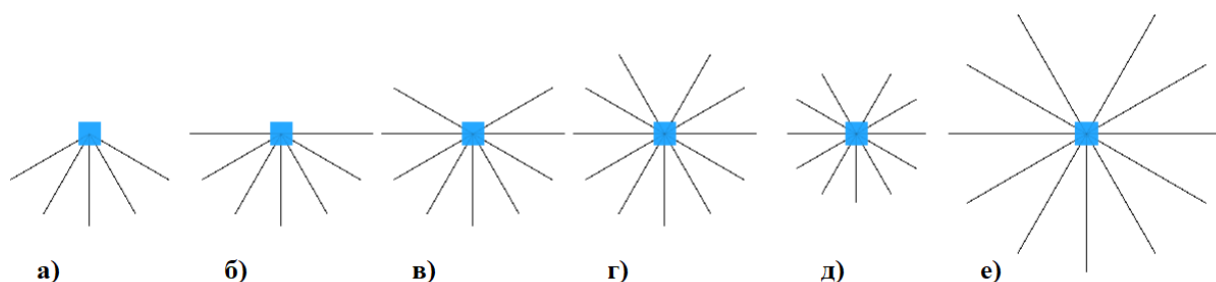


Рисунок 15. Различные конфигурации сенсоров интеллектуальных агентов (а — 120 градусов, б — 180 градусов, в — 240 градусов, г — 300 градусов, д — 300 градусов, максимальная длина луча  $L$  уменьшена на 25%, е — 300 градусов, максимальная длина луча  $L$  увеличена на 50%)

Сконфигурированные искусственные нейронные сети для алгоритмов SAC и PPO имеют по 2 скрытых полносвязных слоя прямого распространения,

имеющих по 256 нейронов. Период обучения агентов составляет 2.5 миллиона эпох. Результаты обучения интеллектуальных агентов с различными конфигурациями сенсоров представлены в таблице 1.

Таблица 1. Результаты обучения интеллектуальных агентов

Конфигурации сенсоров	SAC		PPO	
	Среднее вознаграждение	Функция потерь	Среднее вознаграждение	Функция потерь
а — 120 градусов	2.63	9.95e-4	4.908	0.389
б — 180 градусов	3.012	2.81e-3	4.918	0.307
в — 240 градусов	3.917	3.39e-3	4.981	0.019
г — 300 градусов	4.742	3.97e-3	4.986	0.015
д — 300 градусов, L -25%	1.278	1.29e-3	4.872	0.345
е — 300 градусов, L+50%	4.979	3.39e-4	4.985	0.014

Среднее вознаграждение и функция потерь высчитываются из последних 50000 эпох симуляции. Среднее вознаграждение характеризует то, насколько хорошо выполняется поставленная задача, и может варьироваться от -1 до 5. Из таблицы 1 видно, что с увеличением угла обзора сенсоров агенты выполняют задачу быстрее, и допускают меньше ошибок, однако алгоритм обучения PPO способен найти сложную последовательность решений, которая приводит к хорошим результатам даже в условиях неполноты предоставляемой информации.

Функция потерь характеризует то, как хорошо агент способен предсказывать последствия своих действий. В алгоритме PPO заметна зависимость между предоставляемой информацией и функцией потерь, на основе которой можно подбирать типы сенсоров для различных задач. В алгоритме SAC на основе функции потерь возможен анализ результатов обучения только при достаточной полноте предоставляемой информации. Из всех проведённых опытов, только в последнем функция потерь алгоритма SAC в процессе обучения имела характерную тенденцию роста, а затем уменьшения. Все остальные конфигурации приводили к хаотичным скачкам показателя на каждом этапе обучения.

Стоит отметить, что оба алгоритма значительно теряют эффективность при уменьшении максимального измеряемого лучом расстояния, а при

увеличении данного показателя алгоритм SAC получает достаточно информации для эффективного пространственного восприятия сцены и поиска эффективного решения задачи.

Проанализировав полученные результаты были сделаны следующие выводы:

1. в условиях неполноты или нарочной минимизации входных данных, алгоритм PPO показывает лучшие результаты за более короткое время обучения, нежели SAC;

2. алгоритм SAC в условиях неполноты предоставляемой информации не способен обнаружить сложные, многоуровневые решения задач, найденных алгоритмом PPO;

3. на основе функции потерь алгоритма PPO можно судить о полноте предоставляемой информации. Алгоритм SAC такой возможности не предоставляет.

### **3.5 Модель манипулятора**

Спроектированная модель для управления агентом представляет собой манипулятор с 7 степенями свободы (рисунок 16). Учитывая функцию захвата и бездействие, это даёт выбор из 17 вариантов действий в каждый момент времени.

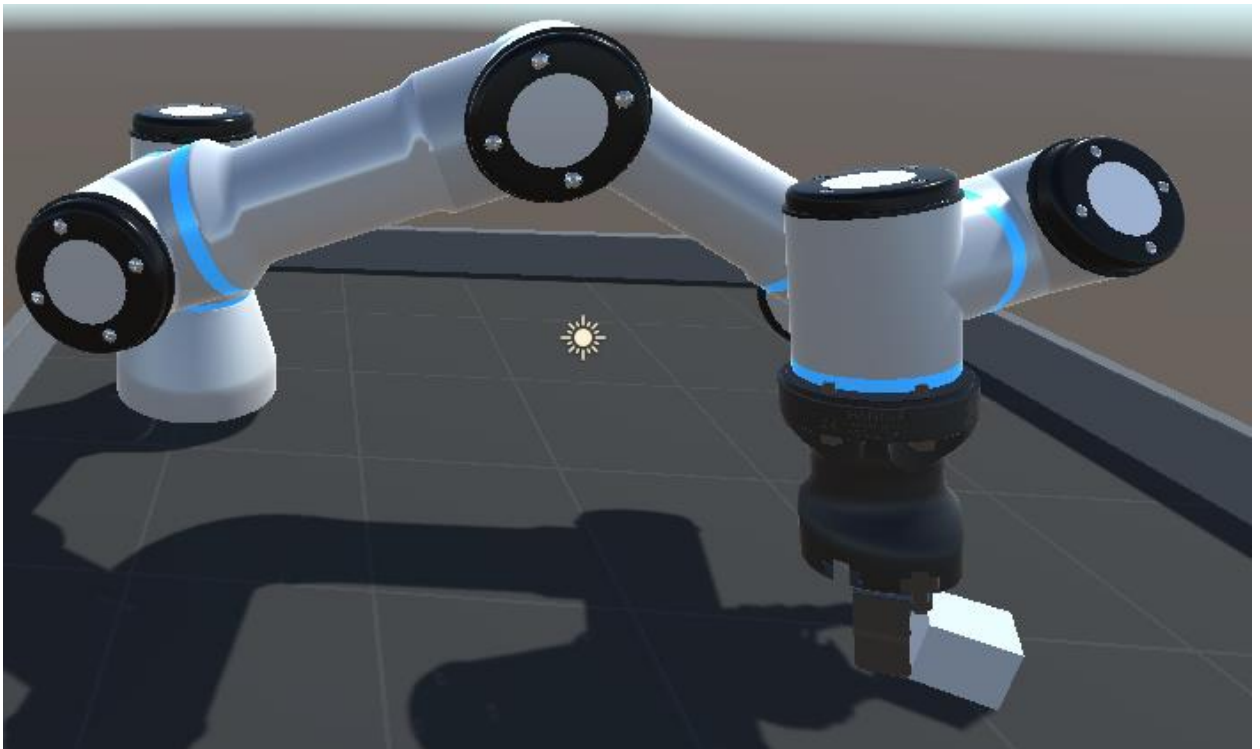


Рисунок 16. Модель манипулятора

Данная модель является аналогом реального манипулятора от компании Universal Robotics (рисунок 17). Эта компания занимается производством и продажей роботов для малого и среднего бизнеса, и позиционирует свою продукцию как универсальных помощников в автоматизации различных производственных процессов: от сборки до покраски, от закручивания винтов до наклеивания этикеток, от упаковки до полировки, от литья под давлением до сварки [21].



Рисунок 17. Роботы манипуляторы от компании universal robots[21]

Конфигурация манипулятора от компании Universal Robotics используется командой AVATRINA на соревновании \$10M ANA Avatar

XPRIZE в разработке своего робота [22]. Данная команда оказалась одной из 38 полуфиналистов данного соревнования, разработав аватар под названием TRINA (рисунок 16).

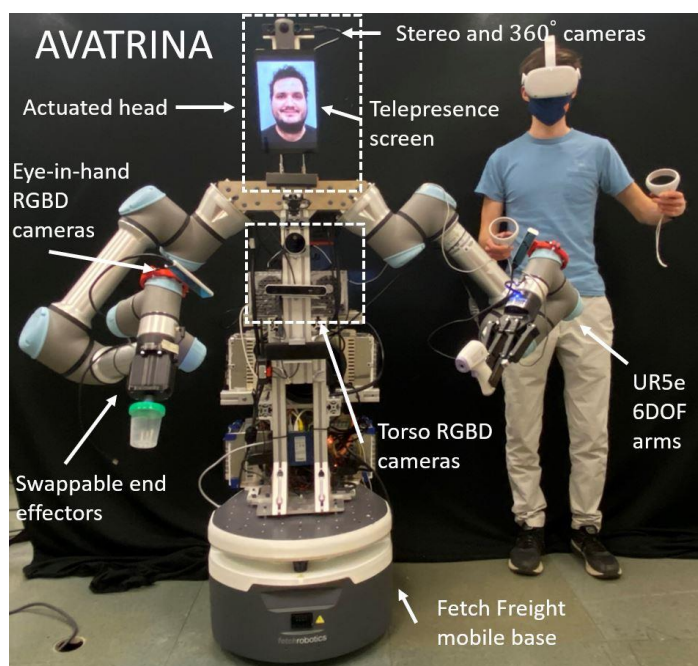


Рисунок 18. Робот TRINA от команды AVATRINA [22]

Основная цель робота: передавать присутствие человека в удаленное место в реальном времени. Для этого он оснащён множеством камер, датчиками глубины, и несколькими манипуляторами. Однако управляемость роботизированными руками таких аватаров затруднена из-за большего числа суставов, что видно на демонстрациях работы на момент написания данной магистерской диссертации. Решить данную проблему можно применив ИНС как посредника между оператором и аватаром.

### 3.6 Результаты обучения по захвату манипулятором объекта

#### 3.6.1. Описание обучающей среды

Смоделированная среда изображена на рисунке 19. Агент управляет роботизированным манипулятором, расположенным на краю выделенной зоны симуляции. Основная цель агента — это захват объекта, являющегося белым кубом. По краям зоны симуляции расположены бортики для

исключения случаев сброса объекта или вывода за края досягаемости манипулятора в процессе обучения.

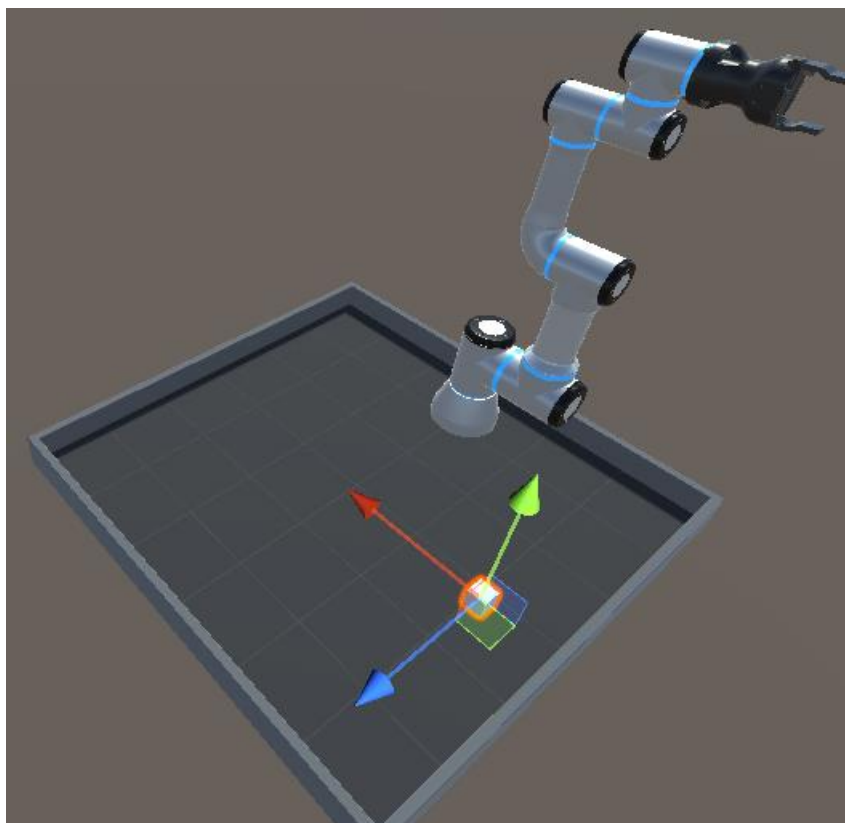


Рисунок 19. Смоделированная среда для захвата объекта

Длительность симуляции ограничена 10000 эпох. В случае, если агент не выполняет поставленную задачу за отведённый период, моделируемая зона перезапускается. При каждом перезапуске манипулятор возвращается в исходное состояние, объект для захвата обнуляет параметры скорости и перемещается в случайную точку зоны симуляции.

Агент выполняет действия на основе информации об его текущем состоянии:

- угол поворота между всеми суставами;
- позиции сочленения относительно основания манипулятора;
- векторы направления отдельных суставов к оси  $X$  в мировом пространстве координат;
- векторы направления отдельных суставов к оси  $Z$  в мировом пространстве координат.

Для взаимодействия манипулятора с объектом, агент получает дополнительную информацию:

- вектор расстояния от основания манипулятора до объекта;
- вектор расстояния от основания манипулятора до хватательного механизма;
- вектор расстояния от хватательного механизма до объекта.

Для того, чтобы агент лучше просчитывал сложные действия и их последствия, в каждый момент симуляции помимо актуальной информации на искусственную нейронную сеть подаются предыдущие данные за последние 3 эпохи. В итоге, минимальный размер входного слоя ИНС насчитывает 237 нейронов.

Помимо вышеописанных данных, дополнительно использовались датчики расстояния, расположенные на хватательном механизме (рисунок 20). Датчики выпускают веер из семи лучей под углом в 60 градусов с ограниченным максимальным расстоянием. При попадании луча, препятствие классифицируется. Расстояние до препятствия возвращается на нейрон, отвечающий за классифицируемый объект.

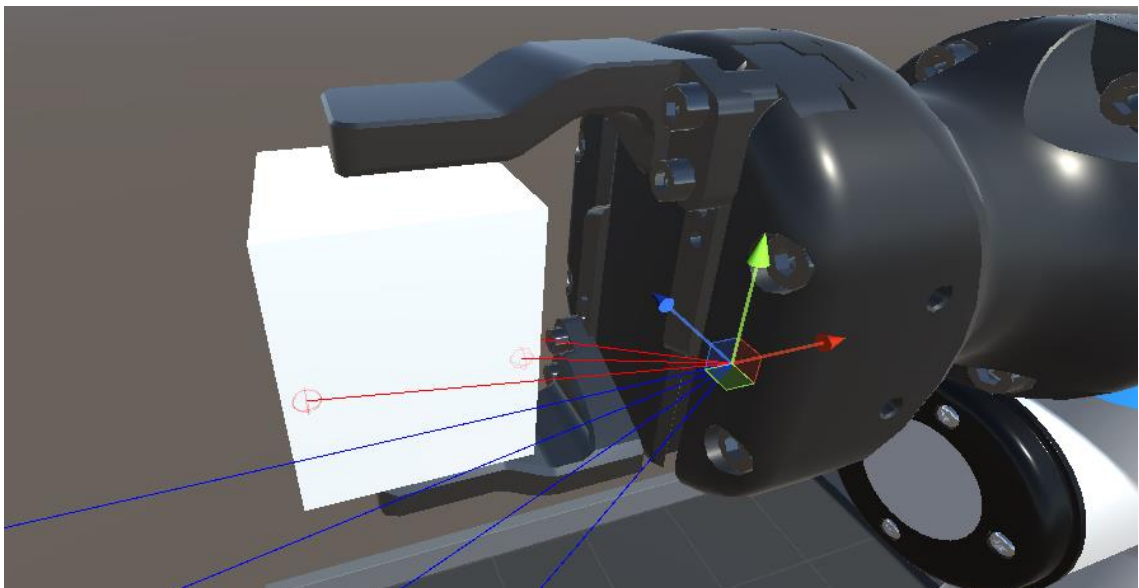


Рисунок 20. Датчики расстояния



Задача поднятия предмета манипулятором для обучения ИНС является довольно не тривиальной с точки зрения конфигурации функции ценности. В отличие от задачи, рассмотренной в разделе 3.4, награды за итоговый результат будет недостаточно по причине необходимости выполнения большой последовательности действий. Это вынуждает добавлять промежуточные награды и штрафы, между которыми необходимо соблюдать баланс для достижения качественного и быстрого обучения.

В ходе выполнения работы были сконфигурированы и обучены более сотни ИНС. Целью данных экспериментов — поиск оптимальных гиперпараметров для функции ценности, анализ их влияния на процесс обучения и построение итоговой модели по поднятию объекта.

Во всех проведённых экспериментах использовалась одинаковая архитектура ИНС, состоящая из двух скрытых слоёв, по 256 нейронов в каждом. Выходной слой состоит из 17 нейронов, характеризующих различные действия манипулятора. Размер входного слоя зависит от количества используемых датчиков, и изменяется в различных экспериментах.

### **3.6.2. Результаты обучения**

Одним из самых важных факторов обучения с подкреплением является функция вознаграждения. От функции вознаграждения зависит не только итоговый результат, но и скорость сходимости алгоритма обучения ИНС. Чем сложнее задача, которой необходимо обучить агентов, тем запутаннее будет итоговая функция вознаграждений.

В задаче по захвату объекта манипулятором использовались следующие характеристики для составления функции вознаграждения:

- награда за захват объекта;
- штраф за время выполнения задачи;
- штраф за дистанцию от захвата до объекта;

- награда за поддержание манипулятором вертикального положения.

Между всеми вышеперечисленными характеристиками необходимо соблюдать баланс. Награда за захват и штраф за время выполнения являются константными характеристиками, и не меняются в процессе симуляции. Для поиска верных коэффициентов между штрафом за дистанцию от захвата до объекта и вознаграждением за поддержание манипулятором вертикального положения была проведена серия тестов. Значение используемых коэффициентов представлено в таблице 2.

Таблица 2. Значение коэффициента в функции вознаграждения различных ИНС

Название ИНС	Коэффициент к поддержанию манипулятором вертикального положения	Коэффициент к штрафу за дистанцию от захвата до объекта	Награда за захват объекта	Штраф за время выполнения задачи
ppo_roboARM_27	1/50	1/10	100	-50
ppo_roboARM_28	1/70			
ppo_roboARM_29	1/90			
ppo_roboARM_30	1/110			

На рисунке 21 продемонстрирован график получаемого вознаграждения в процессе обучения искусственных нейронных сетей, конфигурации которых описаны в таблице 2.

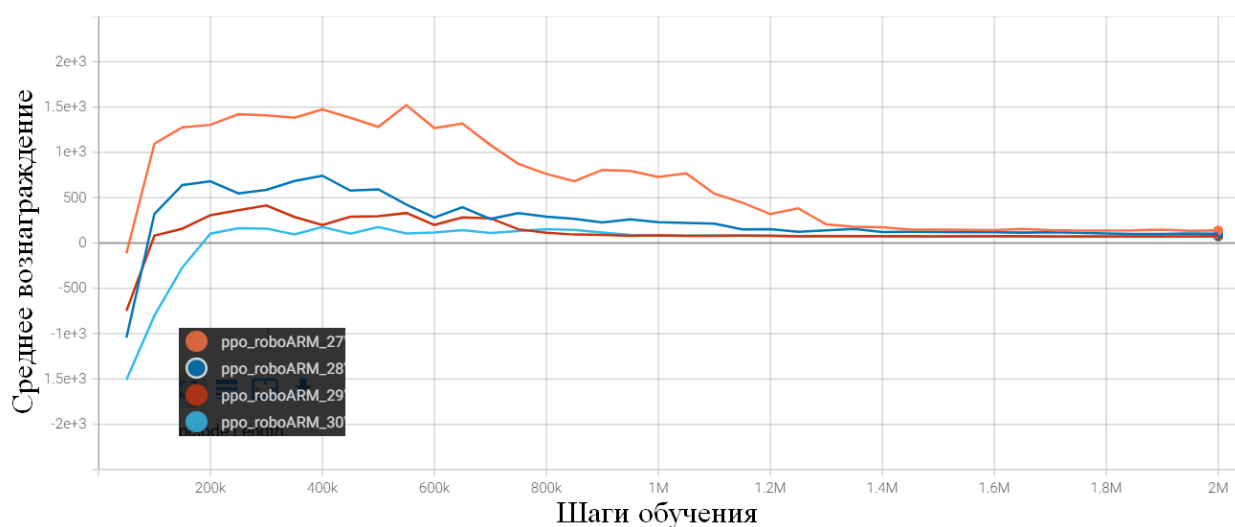


Рисунок 21. Получаемые вознаграждения

Из графиков видно, что все сети успешно обучились захватывать объект. Однако скорость сходимости функций у всех разная. На рисунке 21 видно, что вознаграждение за симуляцию у трёх ИНС в начале обучения выше, чем в конце. Это связано с тем, что агент достаточно быстро научился сокращать расстояние между хватательным механизмом и объектом, что сильно снизило штраф за расстояние. Однако вознаграждение за поддержание манипулятором вертикального положения осталось, и его значение в сумме выше, чем награда за захват. Данную проблему можно решить, увеличив награду за выполнение задачи, или уменьшив все остальные вознаграждения.

На рисунке 22 продемонстрирован график длительности симуляций в процессе обучения искусственных нейронных сетей, конфигурации которых описаны в таблице 2. Чем меньше время симуляции, тем быстрее агент выполняет поставленную задачу — захватывает объект. На графике в период между 600 000 и 1 200 000 шагом обучения видно, что с уменьшением значимости характеристики за поддержание манипулятором вертикального положения, скорость сходимости алгоритма увеличивается, и агент быстрее обучается захватывать объект. Однако в ИНС «ppo\_roboARM\_30» скорость сходимости снова возросла. Это говорит о том, что в ИНС «ppo\_roboARM\_29» достигнут баланс между гиперпараметрами функции вознаграждения.

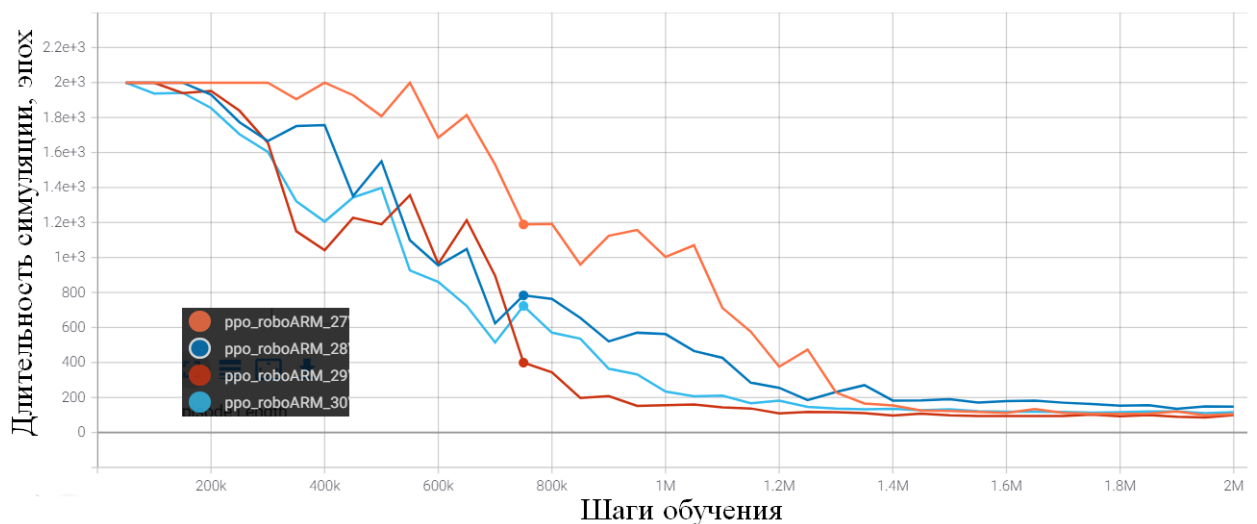


Рисунок 22. Время работы симуляции

В ходе выполнения данных экспериментов были обнаружены несколько проблем:

- 1) При повороте начального положения манипулятора на 180 градусов, агент теряет способность захватывать объекты. Это связано с тем, что все получаемые обучающие данные были смоделированы, когда объекты генерировались в заданной области относительно манипулятора. Функционально, манипулятор способен поворачиваться и поднимать объекты вокруг себя, но из-за отсутствия подобного опыта, агент не способен этого сделать.
- 2) Довольно часто в процессе обучения, даже с корректными гиперпараметрами функции вознаграждения, возникает ситуация, при которой агент предпочитает закрывать хватательный механизм в начале симуляции. Это приводит к неспособности захватить объект.

Для решения первой проблемы была переработана моделируемая среда. Начальное положение манипулятора смещено к центру зоны симуляции (рисунок 23). Вторая проблема решена посредством введения штрафа за ложный захват, и добавлением ко входной информации для ИНС данных о положении захвата — открыт или закрыт. Данные изменения снизили скорость обучения, так как пространство возможных положений объекта относительно манипулятора сильно возросло.

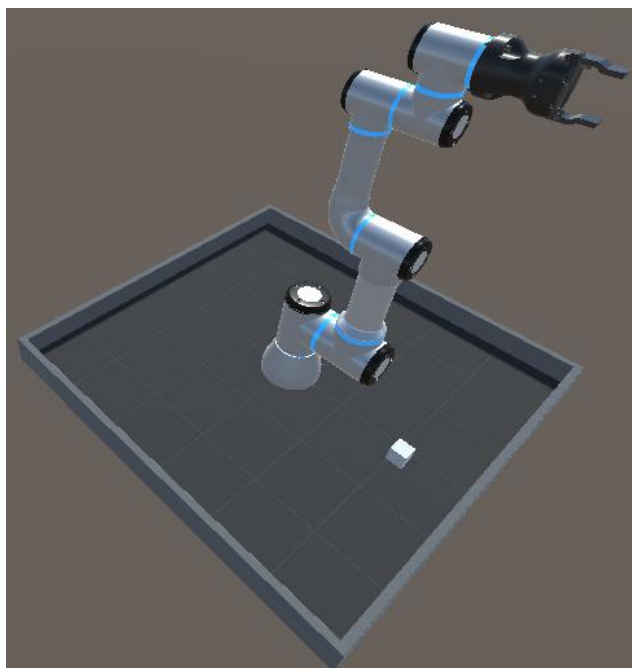


Рисунок 23. Смещение манипулятора в центр зоны симуляции

Была проведена серия тестов по обучения интеллектуальных агентов алгоритмами Soft Actor-Critic (SAC) и Proximal Policy Optimization (PPO). Лучшие из полученных результатов обучения представлены на рисунке 24.

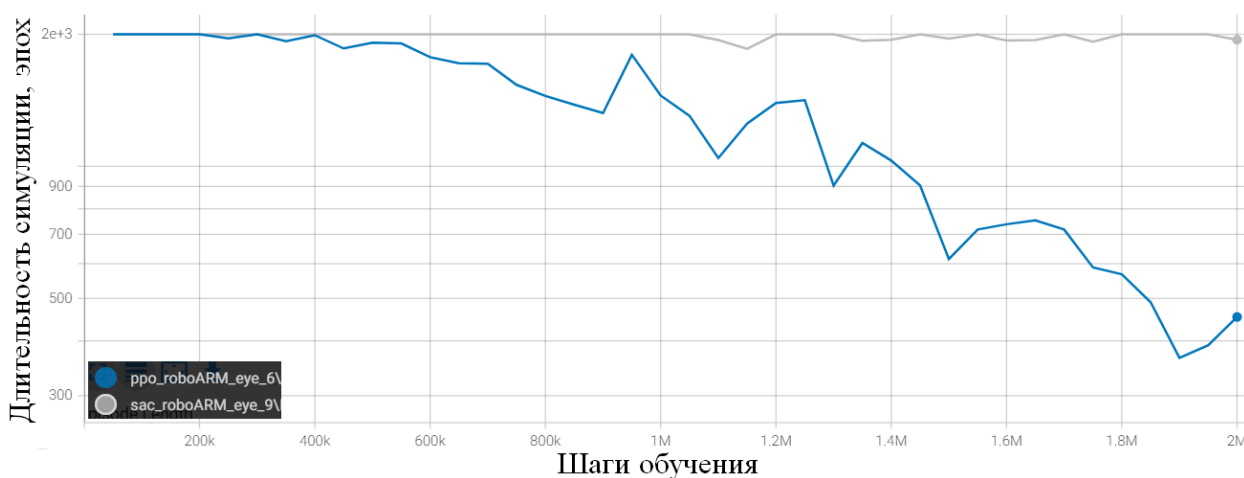


Рисунок 24. Результаты обучения ИНС алгоритмами SAC и PPO

На рисунке 24 продемонстрирован график длительности симуляций в процессе обучения искусственных нейронных сетей алгоритмами SAC и PPO. На основе полученных данных был сделан вывод, что алгоритм SAC не способен решить поставленную задачу на основе предоставляемой ему информации. Основной причиной может служить нехватка исходных данных для прогнозирования лучшей линии поведения, что вынуждает агентов искать

более сложные зависимости. Дальнейшие опыты проводились исключительно с алгоритмом PPO.

Зависимость качества обучения от количества используемых датчиков расстояния продемонстрировано на рисунке 25. На данном графике показаны лучшие из полученных результатов для каждой конфигурации. Количество датчиков в каждой ИНС отмечено в таблице 3.

Таблица 3. Конфигурации ИНС по количеству датчиков

Искусственная нейронная сеть	Количество датчиков
ppo_roboARM_not_eye_1	0
ppo_roboARM_eye_2	2
ppo_roboARM_eye_102	4

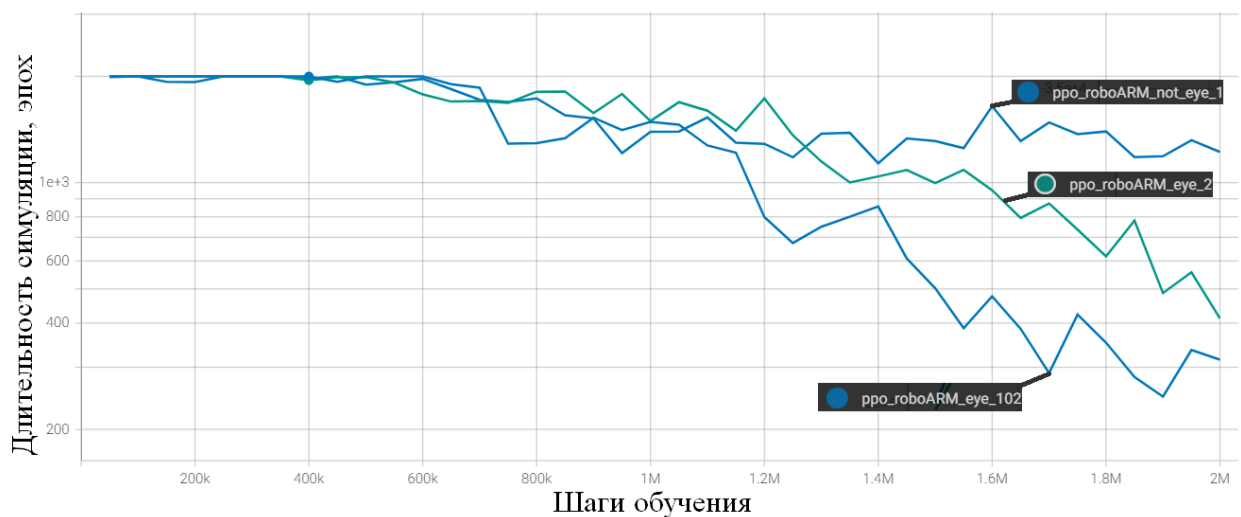


Рисунок 25. Зависимость длительности симуляции от количества используемых датчиков

Как видно из рисунка 25, агент, получающий информацию с 4х датчиков, обучается быстрее остальных. На втором месте стоит агент с двумя датчиками.

Проанализировав результаты, можно сделать вывод, что чем большей полезной предоставляемой информацией оперирует интеллектуальный агент, тем быстрее он обучается выполнять поставленную задачу.

### 3.7 Результаты обучения по поднятию объекта манипулятором

В процессе разработки, задача по захвату объекта была дополнена необходимостью его поднятия на определённую высоту. В функцию вознаграждения был добавлен гиперпараметр, отвечающий за начисление очков за высоту поднятого объекта. Новый интеллектуальный агент имеет название «ppo\_roboARM\_eye\_132», и использует предобученную ИНС «ppo\_roboARM\_eye\_102», конфигурация и задачи которой раскрыты в предыдущем разделе. Графики получаемого вознаграждения в процессе обучения представлены на рисунке 26.

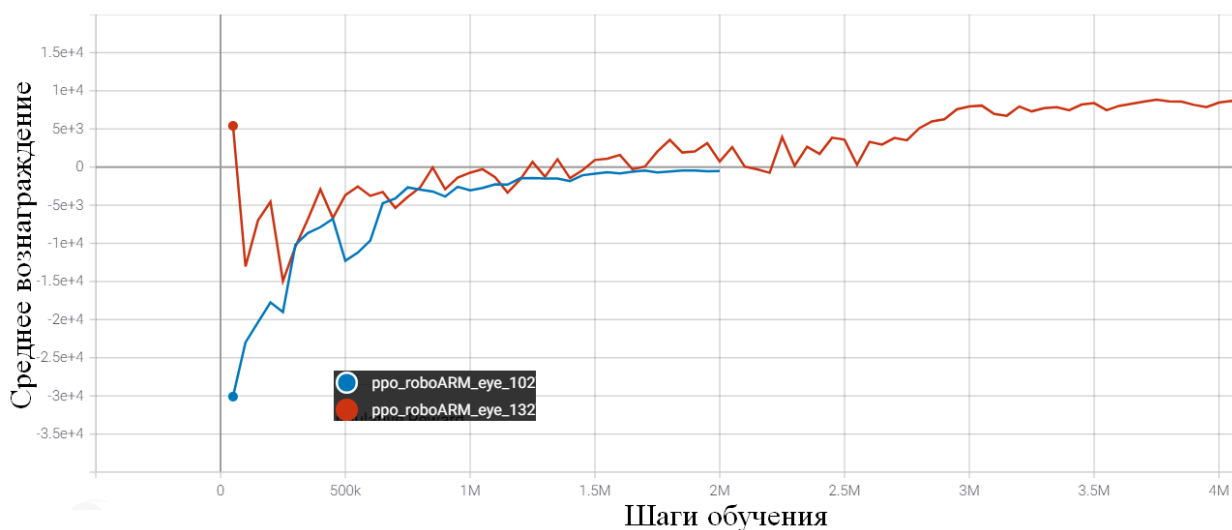


Рисунок 26. Получаемое вознаграждения за поднятие объекта

На рисунке 26 видно, что агент «ppo\_roboARM\_eye\_132» на первых шагах обучения получил высокую награду за выполнение задачи, но в последствии величина вознаграждения упала ниже, чем ИНС «ppo\_roboARM\_eye\_102» получала в конце своего обучения. Основной причиной такого падения стало несвоевременное выполнение задачи. Агент получил большое вознаграждение, но не имел опыта исследования окружающей среды — только рефлексы по захвату объектов, переданные с коэффициентами межнейронных связей агента «ppo\_roboARM\_eye\_102». Итогом стали длительные исследования смоделированной среды с целью повторного получения большого вознаграждения.

Рассматривая тенденцию получаемого вознаграждения видно, что в период приблизительно с 50 000 до миллиона шагов обучения агент «ppo\_roboARM\_eye\_132» исследовал окружающую среду и редко захватывал объект. В период с миллиона до двух с половиной миллионов агент захватывал объект и приподнимал на небольшую высоту. В период с двух с половиной до трёх миллионов шагов происходит качественный скачок, агент начинает регулярно и быстро поднимать объект на заданную высоту.

Подводя итоги эксперимента, можно сделать вывод, что использование предобученных агентов позволяет переобучить нейронную сеть для выполнения новых, более сложных и многоступенчатых задач, но на перестройку имеющихся межнейронных связей, учитывая необходимость сохранения имеющихся возможностей, уходит некоторое время.

### **3.8 Выводы по разделу**

В данном разделе были реализованы и протестированы алгоритмы обучения интеллектуальных агентов PPO и SAC с различными конфигурациями получаемой информации от окружающей среды. В результате экспериментов было обнаружено, что алгоритм PPO лучше справляется с обнаружением сложных, многоуровневых последовательностей для решения поставленной задачи.

Была спроектирована виртуальная среда по моделированию работы манипулятора. Для лучшего анализа получаемых результатов, задача по поднятию объектов была разделена на следующие подзадачи: захват объекта и поднятие объекта. В ходе проведения большего количества опытов по обучению интеллектуальных агентов задаче по захвату объекта, ИНС обучилась быстро, и успешно захватывать объекты в 100% симуляций. Скорость выполнения задачи в среднем составляла 300 эпох. Каждая эпоха характеризуется выполнением выбранного действия в течении определённого времени.



После анализа полученных результатов были выявлены и исправлены такие ошибки, как:

- дисбаланс гиперпараметров функции ценности, приводящий к снижению скорости обучения агентов;
- ошибка отсутствия штрафа за ложный захват объектов, с некоторой вероятностью способная привести модель в состояние с постоянным закрытым механизмом захвата, при котором модель теряла способность обучаться;
- ошибка смоделированной среды обучения, при которой объекты появлялись с одной и той же стороны относительно манипулятора, что приводило к неспособности агента обобщить информацию для захвата объектов с других направлений.

Для обучения агента задаче по поднятию объектов была использована предобученная ИНС, полученная на предыдущем этапе. Данный подход показал свою эффективность с точки зрения прозрачности получаемых результатов, так как позволил решать и анализировать поставленную задачу поэтапно.

Все эксперименты проводились на ноутбуке с характеристиками:

- Процессор Intel® Core™ i5-9300H
- Графический процессор: NVIDIA GeForce GTX 1650
- ОЗУ 8 Гб (1 поток)

Время обучения интеллектуальных агентов зависит от CPU и GPU ЭВМ, сложности смоделированного пространства, сложности поставленной задачи и конфигурации ИНС. Для обучения агента задаче по захвату объектов при использовании данного оборудования уходит в среднем 2.5 часа. Время обучения задаче по подъёму объектов составляло в среднем 6 часов.

#### **4. Финансовый менеджмент, ресурсоэффективность и ресурсосбережение**

##### **4.1. Оценка коммерческого потенциала и перспективности проведения научных исследований с позиции ресурсоэффективности и ресурсосбережения**

###### **4.1.1. Потенциальные потребители результатов исследования**

Машинное обучение, в частности искусственные нейронные сети, не смотря на давнее появление, широкое коммерческое развитие получили относительно недавно. Особенно широко они распространились в области распознавания (голоса, изображения), в области прогнозирования временных рядов, а с недавних пор и в робототехнике.

Крупные компании вроде Google, Amazon, Microsoft, Tesla и Яндекс имеют собственные подразделения и разработки в данной области. Фирмам поменьше приходится искать готовые решения или заказывать собственную разработку. Спрос на нейронные сети высок, т.к. они позволяют автоматизировать многие процессы, которые автоматизировать с помощью обычных алгоритмов сложно и дорого [1].

В данной работе исследуется возможность применения нейронных сетей для автоматизации любых процессов, позволяющих использовать беспилотные роботизированные аппараты. Потенциальными потребителями могут стать поисково-спасательные службы, т.к. группа дронов способна исследовать огромную территорию за малый промежуток времени, космическая промышленность, т.к. управляющий сигнал в зависимости от расстояния может не позволять напрямую управлять роботизированным устройством из-за колоссальных задержек, военная сфера применения, т.к. управляющий беспилотным аппаратом сигнал может глушиться злоумышленником, мусороперерабатывающая сфера, т.к. сортировка отходов является трудоёмким, монотонным и не безопасным процессом и т.д.

#### 4.1.2. Технология QuaD

Идея применения нейронных сетей для управления беспилотными аппаратами возникла давно, но технически это стало возможно лишь недавно. В настоящий момент широкое распространения в данной области получила только компания Tesla с её автопилотом для автомобильного транспорта, но данная задача требует гигантских вычислительных мощностей и более сложные обучающие алгоритмы.

Суть данной работы заключается в разработке алгоритма обучения нейронных сетей для выполнения относительно простых задач по типу обхода агентом препятствий, или управление механическим хватательным устройством. Данные алгоритмы существуют и применяются в настоящее время, но их реализация основана на жестких, линейных алгоритмах без применения нейронных сетей.

Рассмотрим основные критерии оценки разрабатываемого:

**Надёжность** характеризует степень отказоустойчивости алгоритма и способность выполнения поставленных задач без ошибок.

**Масштабируемость** характеризует возможность интеграции дополнительного функционала, не предусмотренного в изначальном проекте.

**Гибкость** характеризует возможность изменить поставленную алгоритму задачу с минимальными прилагаемыми людскими ресурсами.

**Потребность в ресурсах памяти** характеризует необходимость ОЗУ для работы алгоритма.

**Эффективность** характеризует возможность оптимального, с минимальными временными и ресурсными затратами, выполнять поставленные задачи алгоритмом.

**Простота эксплуатации** характеризует сложность работы с разработанным алгоритмом.

**Потребность в мощности ЭВМ** характеризует необходимость в вычислительных мощностях для обучения нейронных сетей.

**Скорость работы** характеризует то, как быстро алгоритм способен обучать ИНС выполнению поставленных задач.

По технологии QuaD построим оценочную карту нашей новой разработки (Таблица 4).

Таблица 4. Оценочная карта технического решения

Критерии оценки	Вес критерия	Баллы	Максимальный балл	Относительное значение	Среднее взвешенное
1	2	3	4	5	6
<b>Показатели оценки качества разработки</b>					
Надёжность	0,07	75	100	0,75	0,0525
Масштабируемость	0,12	80	100	0,8	0,096
Гибкость	0,12	100	100	1	0,12
Потребность в ресурсах памяти	0,07	80	100	0,8	0,056
Эффективность	0,12	100	100	1	0,12
Простота эксплуатации	0,03	35	100	0,35	0,0105
Потребность в мощности ЭВМ	0,08	40	100	0,4	0,032
Скорость работы	0,08	65	100	0,65	0,052
<b>Показатели оценки коммерческого потенциала разработки</b>					
Уровень проникновения на рынок	0,03	50	100	0,5	0,015
Конкурентоспособность	0,1	85	100	0,85	0,085
Послепродажное обслуживание	0,08	70	100	0,7	0,056
Готовность продукта	0,1	30	100	0,3	0,03
Итого:	1	810	1200	0,675	0,725

Оценка качества и перспективности по QuaD следующим образом (1).

$$P_{cp} = \sum B_i B_i = 0.725 \quad (1)$$

Исходя из этой цифры, делаем вывод, что перспективность данного проекта выше среднего.

#### 4.1.3. SWOT-анализ

SWOT-анализ дает возможность оценить сильные и слабые стороны, возможности и угрозы проекта методом стратегического планирования.

Таблица 5. SWOT-анализ проекта

Внешние факторы	Внутренние факторы		
		Сильные стороны проекта:	Слабые стороны проекта:
		С1. Проект востребован С2. Аналогов очень мало С3. Много отраслей для применения	Сл1. Сложность разработки Сл2. Тестирование занимает много времени Сл3. Необходимы большие вычислительные ресурсы Сл4. Сложно отслеживать ошибки Сл5. Необходим высококвалифицированный персонал
	Возможности:	Проект востребован, а конкуренции мало. Существует огромный рынок для применения технологии.	Время на тестирование и потребности в вычислительных ресурсах со временем будут меньше влиять на сложность проекта т.к. развитие информационных и облачных технологий, а также вычислительных систем постоянно упрощают решение вычислительных задач.
	Угрозы:	Востребованность проекта будет привлекать множество конкурентов. Большое количество отраслей служат защитой компании от убытков при вмешательстве гос. в какую-либо из них.	Повышение цен на комп. оборудование может сильно повысить стоимость и время разработок. Цены также растут из-за ослабления курса рубля. Для уменьшения данных рисков можно хранить финансы в долларах(евро), а также закупить оборудования для разработок.

Из анализа можно сделать вывод, что актуальность данного исследования растёт, и будет расти ещё долгое время. Все возможные

сдерживающие факторы нивелируются быстрым развитием вычислительных систем и информационных технологий, что открывает путь для маленьких и средних компаний.

## **4.2. Планирование научно-исследовательских работ**

Данный раздел необходим для планирования будущих работ, оценки каждого участка и определения его продолжительности, а также прогнозирование итоговой продолжительности проекта. При планировании таких работ необходимо оптимально планировать занятость каждого из участников проекта.

### **4.2.1. Структура работ в рамках научного исследования**

Процесс проведения работ в рамках научного исследования имеет определённую структуру и может быть разбит на этапы, которые включают в себя список работ и их исполнителей. Перечень этапов, работ и распределение исполнителей приведены ниже (Таблица 6).

Таблица 6. Перечень этапов, работ и распределение исполнителей

№ работы	Наименование работы	Исполнители работы
1	Выбор научного руководителя магистерской работы	Пантюхин Е.И.
2	Составление и утверждение темы магистерской работы	Иванова Ю.А. Пантюхин Е.И.
3	Составление календарного плана-графика выполнения магистерской работы	Иванова Ю.А.
4	Подбор и изучение литературы по теме магистерской работы	Пантюхин Е.И.
5	Анализ предметной области	Пантюхин Е.И.
6	Проектирование структуры модели	Пантюхин Е.И.
7	Программирование спроектированных моделей	Пантюхин Е.И.
8	Тестирование запрограммированных моделей	Пантюхин Е.И.

9	Анализ результатов	Пантюхин Е.И.
10	Согласование выполненной работы с научным руководителем	Иванова Ю.А. Пантюхин Е.И.
11	Выполнение других частей работы (финансовый менеджмент, социальная ответственность)	Пантюхин Е.И.
12	Подведение итогов, оформление работы	Пантюхин Е.И.

#### 4.2.2. Определение трудоемкости выполнения работ

Для определения трудоёмкости работ, необходимо оценить минимальное и максимальное затраченное на работу время. Произведём расчёт ожидаемой трудоёмкости это с помощью формулы (2).

$$t_{ож\ i} = \frac{3t_{min\ i} + 2t_{max\ i}}{5} \quad (2)$$

Учитывая, что исполнитель один, ускорить работу за счёт распараллеливания не представляется возможным.

#### 4.2.3. Разработка графика проведения научного исследования

Построим таблицу временных показателей научного исследования (Таблица 7). Согласно производственному календарю (для 6-дневной рабочей недели) в 2021 году 365 календарных дней, 299 рабочих дней, 66 выходных/праздничных дней [23]. Для расчёта длительности работ в календарных днях рассчитаем коэффициент календарности (3)

$$T_{кал} = \frac{T_{кал}}{T_{кал} - T_{вых} - T_{пр}} = \frac{365}{365 - 66} = 1,22 \quad (3)$$

Таблица 7. Временные показатели научного исследования

Наименование работы	Исполнители работы	Трудоемкость работ, чел-дни			Длительность работ, дни	
		tmin	tmax	тож	Тр	Тк
Выбор научного руководителя магистерской работы	Пантюхин Е.И.	1	2	1,4	1	2

Составление и утверждение темы магистерской работы	Пантюхин Е.И.	1	2	1,4	1	1
	Иванова Ю.А.	1	2	1,4	1	1
Составление календарного плана-графика выполнения магистерской работы	Иванова Ю.А.	1	3	1,8	1	2
Подбор и изучение литературы по теме магистерской работы	Пантюхин Е.И.	2	5	3,2	3	4
Анализ предметной области	Пантюхин Е.И.	2	5	3,2	3	4
Проектирование структуры модели	Пантюхин Е.И.	3	6	4,2	4	5
Программирование спроектированных моделей	Пантюхин Е.И.	15	30	21	21	26
Тестирование запрограммированных моделей	Пантюхин Е.И.	20	40	28	28	34
Анализ результатов	Пантюхин Е.И.	5	10	7	7	9
Согласование выполненной работы с научным руководителем	Пантюхин Е.И.	5	8	6,2	3	4
	Иванова Ю.А.	5	8	6,2	3	4
Выполнение других частей работы (финансовый менеджмент, социальная ответственность)	Пантюхин Е.И.	5	10	7	7	9
Подведение итогов, оформление работы	Пантюхин Е.И.	7	10	8,2	4	5
	Иванова Ю.А.	7	10	8,2	4	5



На основе построенной выше таблицы, построим диаграмму Ганта (Рисунок 27).

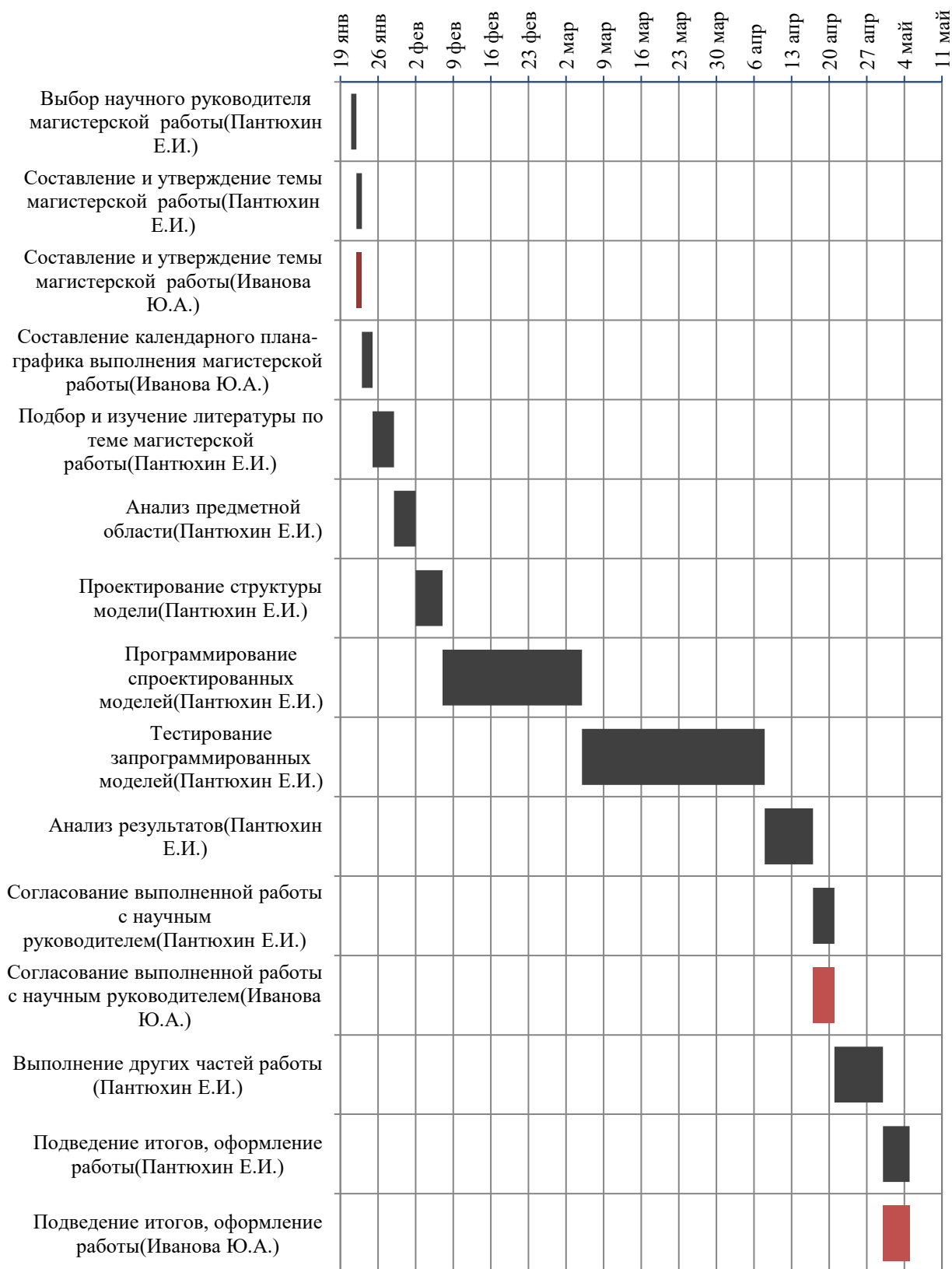


Рисунок 27. Диаграмма Ганта

#### **4.2.4. Бюджет научно-технического исследования**

Затраты на создание проекта состоят из всех расходов необходимых для реализации комплекса работ, составляющих содержание данного исследования. Сметная стоимость формируется по следующим статьям затрат:

- расходы на электроэнергию (без освещения);
- амортизационные отчисления;
- командировочные расходы;
- материалы и покупные изделия;
- заработная плата;
- социальный налог;
- оплата услуг связи;
- арендная плата за пользование имуществом;
- прочие услуги (сторонних организаций);
- прочие (накладные расходы) расходы.

Статьи расходов такие как услуги сторонних организаций, накладные расходы, командировочные расходы, также могут не учитываться, т.к. по этим статьям за период разработки затрат не было.

Проведем расчет затрат на разработку проекта по следующим статьям расходов: затраты на заработную плату участников и социальный налог.

##### **Основная заработная плата исполнителей темы**

При разработке ПО основной статьёй расходов является заработная плата специалистов. Поэтому для расчёта расходов на НТИ необходимо посчитать оклад специалистов за время исследования. Расходы определяются по следующей формуле (4).

$$З_{\pi} = З_{\text{осн}} + З_{\text{доп}} \quad (4)$$

Где:

$З_{\text{осн}}$  – основная заработная плата, руб.

$З_{\text{доп}}$  – дополнительная заработная плата, руб.

Основная заработная плата высчитывается по формуле (5).

$$З_{\text{осн}} = З_{\text{дн}} * Т_{\text{р}} * К_{\text{р}}, \quad (5)$$

Где:

З<sub>дн</sub> – среднедневная заработная плата, руб;

Т<sub>р</sub> – продолжительность работ, выполняемых работником, раб. дни;

К<sub>р</sub> – районный коэффициент (1,3);

Среднедневная заработная плата высчитывается по формуле (6).

$$З_{\text{дн}} = \frac{З_{\text{м}} \times М}{F_{\text{д}}} \quad (6)$$

Где:

З<sub>м</sub> – месячный оклад работника, руб.,

М – количество месяцев работы без отпуска в течение года равное 10,4 месяца для шестидневного рабочего дня.

F<sub>д</sub> – действительный годовой фонд рабочего времени персонала, раб. дн. (Таблица 8).

Таблица 8. Баланс рабочего времени

<b>Показатели рабочего времени</b>	<b>Руководитель</b>	<b>Инженер</b>
Календарное число дней	365	365
Количество нерабочих дней (выходные дни + праздничные дни)	63	96
Потери рабочего времени (отпуск + невыходы по болезни)	56	24
Действительный годовой фонд рабочего времени	254	217

Расчёт основной заработной платы приведён ниже, З<sub>осн</sub> рассчитана с учётом районного коэффициента (1.3) (Таблица 9).

Таблица 9. Расчёт основной заработной платы исполнителей

Исполнители	З <sub>б</sub> , руб.	k <sub>р</sub>	З <sub>м</sub> , руб	З <sub>дн</sub> , руб.	T <sub>р</sub> , дн.	З <sub>осн</sub> , руб.
Руководитель	35 111,50	1.3	45 645	1869	10	18 689
Инженер	22 695	1.3	29 503	1523	82	124 866
Итого по статье З <sub>осн</sub> :			143 556			

Затраты по дополнительной заработной плате исполнителей учитывают величину предусмотренных Трудовым кодексом РФ доплат за отклонение от нормальных условий труда, а также выплат, связанных с обеспечением гарантий и компенсаций (при исполнении государственных и общественных обязанностей, при совмещении работы с обучением, при предоставлении ежегодного оплачиваемого отпуска и т.д.). Дополнительная заработная плата начисляется по формуле, приведённой ниже (7).

$$З_{\text{доп}} = З_{\text{осн}} * k_{\text{доп}} \quad (7)$$

где  $k_{\text{доп}}$  – коэффициент дополнительной з/п. Принимаем его равным 0.15. После расчёта дополнительной заработной платы получаем следующие цифры (Таблица 10)

Таблица 10. Дополнительная заработная плата исполнителей

Исполнители	З <sub>осн</sub>	К <sub>доп</sub>	З <sub>доп</sub> , руб.
Инженер	124 866	0,15	18 730
Руководитель	18 689	0,15	2803

Теперь необходимо рассчитать отчисления во внебюджетные фонды. Они рассчитываются следующим образом (8).

$$З_{\text{внеб}} = k_{\text{внеб}} * (З_{\text{осн}} + З_{\text{доп}}), \quad (8)$$

где  $k_{\text{внеб}}$  – коэффициент отчислений на уплату во внебюджетные фонды (пенсионный фонд, фонд обязательного медицинского страхования и пр.) Для образовательных и научных учреждений он составляет 30%. Результат вычислений приведён ниже (Таблица 11).

Таблица 11. Отчисления во внебюджетные фонды

Исполнитель	Зосн, руб	Здоп руб	Квнеб	Зстр
Инженер	124 866	18 730	30%	43 079
Руководитель	18 689	2803	30%	6 448
Итого:				49527

Величина затрат научно-исследовательской работы является основой для формирования бюджета затрат проекта, который при формировании договора с заказчиком защищается научной организацией в качестве нижнего предела затрат на разработку научно-технической продукции. На основе предыдущих расчётов, можно вывести итоговый бюджет НТИ (Таблица 12).

Таблица 12. Расчет бюджета затрат НТИ

Наименование	Сумма, руб.	Удельный вес, %
Затраты на основную заработную плату	143 556	58 %
Затраты на дополнительную заработную плату	21 533	9 %
Страховые взносы	49 527	20 %
Накладные расходы	34 339	14 %
Общий бюджет	248 955	100 %

#### **4.3. Определение ресурсной (ресурсосберегающей), финансовой, бюджетной, социальной и экономической эффективности исследования**

Определение эффективности происходит на основе расчета интегрального показателя эффективности научного исследования. Его

нахождение связано с определением двух средневзвешенных величин: финансовой эффективности и ресурсоэффективности.

Интегральный финансовый показатель разработки определяется как:

$$I_{\text{финр}}^{\text{исп.}i} = \frac{\Phi_{pi}}{\Phi_{\text{max}}},$$

где  $I_{\text{финр}}^{\text{исп.}i}$  – интегральный финансовый показатель разработки;

$\Phi_{pi}$  – стоимость  $i$ -го варианта исполнения;

$\Phi_{\text{max}}$  – максимальная стоимость исполнения научно-исследовательского проекта.

$$I_{\Phi}^p = \frac{\Phi_i^p}{\Phi_{\text{max}}} = \frac{248\,955}{300\,000} = 0.83$$

$$I_{\Phi}^a = \frac{\Phi_i^a}{\Phi_{\text{max}}} = \frac{195\,064}{300\,000} = 0.65$$

$$I_{\Phi}^b = \frac{\Phi_i^b}{\Phi_{\text{max}}} = \frac{280\,015}{300\,000} = 0.93$$

Полученная величина интегрального финансового показателя разработки отражает соответствующее численное увеличение бюджета затрат разработки в размах (значение больше единицы), либо соответствующее численное удешевление стоимости разработки в размах (значение меньше единицы, но больше нуля).

Интегральный показатель ресурсоэффективности вариантов исполнения объекта исследования можно определить следующим образом:

$$I_{pi} = \sum a_i \cdot b_i,$$

где  $I_{pi}$  – интегральный показатель ресурсоэффективности для  $i$ -го варианта исполнения разработки;

$a_i$  – весовой коэффициент  $i$ -го варианта исполнения разработки;

$b_i^a, b_i^p$  – балльная оценка  $i$ -го варианта исполнения разработки, устанавливается экспертным путем по выбранной шкале оценивания;

$n$  – число параметров сравнения.

Таблица 13. Сравнительная оценка характеристик вариантов исполнения проекта

Объект исследования	Весовой коэффициент параметра	Текущий	Аналог 1	Аналог 2
Критерии				
Надёжность	0,07	75	50	90
Масштабируемость	0,12	80	80	80
Гибкость	0,12	100	100	100
Потребность в ресурсах памяти	0,07	80	75	80
Эффективность	0,12	100	95	100
Простота эксплуатации	0,03	35	30	45
Потребность в мощности ЭВМ	0,08	40	40	40
Скорость работы	0,08	65	65	75
Уровень проникновения на рынок	0,03	50	40	65
Конкурентоспособность	0,1	85	70	95
Послепродажное обслуживание	0,08	70	60	70
Готовность продукта	0,1	30	30	60
ИТОГО	1	72,5	67,05	79,1

$$I_{p-исп1} = 72,5$$

$$I_{p-исп2} = 67,05$$

$$I_{p-исп3} = 79,1$$

Сравнение значений интегральных показателей эффективности позволяет судить о приемлемости существующего варианта решения поставленной в магистерской диссертации технической задачи с позиции финансовой и ресурсной эффективности.

#### 4.4. Выводы по разделу

В ходе данной работе были рассмотрены потенциальные потребители результатов исследования, так же для анализа конкурентных технических решений. Потенциальными потребителями могут стать любые компании, выпускающие роботизированные аппараты: роботы-пылесосы, квадрокоптеры, подводные аппараты, сортировочные, погрузочные аппараты и т.д. С внедрением исследуемой технологии данные аппараты станет проще программировать для задач, которые сложно и дорого решить иными методами.

Далее был сформирован SWOT-анализ, в котором балы описаны сильные и слабые стороны проекта, в выявлении возможностей и угроз для реализации проекта.

Был сформирован план проекта, представленный на диаграмме Ганта, из которого видно какой исполнитель (студент или руководитель) какой вид работ осуществлял и в течении какого количества дней.

Был рассчитан интегральный финансовый показатель в ходе оценки бюджета затрат трех вариантов исполнения. Наибольший интегральный показатель реализации задачи равен 0.93. Значение меньше единицы, а значит полученная величина показывает численное удешевление стоимости в размах. Рассчитан интегральный показатель ресурсоэффективности для трех вариантов исполнения 72.5, 67.05 и 69.1. Расчет интегрального показателя эффективности для разработки и аналога позволил сравнить их и рассчитать сравнительную эффективность разработки, которая говорит нам о приемлемости существующего варианта решения поставленной в магистерской диссертации технической задачи с позиции финансовой и ресурсной эффективности.



## **5. РАЗДЕЛ «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»**

### **5.1. Введение**

В научно-исследовательской и проектной деятельности необходимо учитывать безопасность труда и окружающей среды. Под понятием «социальная ответственность» понимается состояние рабочего места и помещения, режим трудовой деятельности и обеспечение мероприятий по защите трудящихся в моменты чрезвычайных ситуаций.

### **5.2. Правовые и организационные вопросы обеспечения безопасности**

#### **5.2.1. Правовые нормы трудового законодательства**

В Российской Федерации вопросы, относящиеся к организации и охране труда, регулируются трудовым кодексом [24]. Документ устанавливает следующие правила:

- количество рабочих часов в неделю не должно превышать сорока часов;
- продолжительность рабочего дня перед праздничным или выходным днём уменьшается на один час;
- работникам раз в год предоставляются отпуска продолжительностью 28 дней с сохранением места работы (должности) и среднего заработка;
- в течение рабочего дня работнику должен предоставляться, перерыв не более двух часов и не менее 30 минут, не включаемый в рабочее время;
- работа в выходные дни проводится только с согласия работника.

Также существуют документ, регламентирующий некоторые аспекты работы на персональном компьютере – «Типовая инструкция по охране труда

при работе на персональном компьютере» (ТОИ Р-45-084-01) [25]. Он устанавливает следующие правила:

- Соблюдать режим труда и отдыха в зависимости от продолжительности, вида и категории трудовой деятельности
- Продолжительность непрерывной работы с компьютером без регламентированного перерыва не должна превышать 2-х часов.
- Во время регламентированных перерывов с целью снижения нервно - эмоционального напряжения, утомления зрительного анализатора, устранения влияния гиподинамии и гипокинезии, предотвращения развития позотонического утомления выполнять комплексы упражнений.

#### **5.2.2. Организационные мероприятия при компоновке рабочей зоны**

При компоновки рабочей зоны необходимо учесть то, что основную часть работы инженер проводит перед монитором в неподвижном положении, следовательно, следует рассмотреть ряд вопросов, касающийся компоновки рабочего места с учётом эргономики.

Согласно ГОСТ 12.2.032-78 ССБТ [26] рабочее место для выполнения работ сидя организуют при легкой работе, не требующей свободного передвижения работающего, а также при работе средней тяжести в случаях, обусловленных особенностями технологического процесса. Конструкция рабочего места и взаимное расположение всех его элементов (сиденье, органы управления, средства отображения информации и т.д.) должны соответствовать антропометрическим, физиологическим и психологическим требованиям, а также характеру работы.

Высота рабочего места подбирается исходя из видов работ, которые будут выполняться на производстве, в данном случае работа будет связана с ЭВМ, следовательно, рабочее место будет настраиваться в пределах от 680-

750 мм над уровнем пола. Конструкция регулируемого кресла должна обеспечивать оптимальную рабочую позу для конкретного роста.

Рабочая поверхность должна обеспечивать работу для дисплея, устройства ввода данных и дополнительного оборудования, материалов, а также для кистей рук и локтей пользователя. При работе в позах сидя и стоя также достаточно иметь пространственный зазор между телом пользователя и частями рабочей станции. Также пользователь должен иметь возможность наклонить или повернуть видеодисплей таким образом, чтобы сохранить ненапряженную рабочую позу независимо от высоты уровня глаз. Угол обзора не должен превышать 40 градусов по всей активной площади экрана.

В соответствии с ГОСТ Р ИСО 9241-4-2009 [27] при использовании стандартной съемной клавиатуры, следует учитывать тот факт, что конструкция клавиатуры может влиять на эффективность, результативность работы и удовлетворенность пользователя. Характеристики клавиатуры, которые влияют на эффективность работы, включают в себя: раскладки буквенных и числовых клавиш, лингвистические различия, физические характеристики отдельных клавиш и общую конфигурацию корпуса клавиатуры. Определенные характеристики клавиатуры могут влиять на осанку, принимаемую оператором, работающим с видеотерминалом. Высота основного ряда клавиатуры для оператора должна составлять от 30 до 35 мм. Угол наклона клавиатуры по отношению к оператору – от 50 до 120 и должен быть регулируемым. Также клавиатура должна быть легко установлена на рабочей поверхности, не скользить и не качаться.

При выполнении раздела социальная ответственность были установлены и исследованы на соответствие нормам вредные и опасные факторы при выполнении дипломной работы. По результатам данного исследования были приняты меры по противодействию данным факторам. Также была рассмотрена экологическая безопасность, где было установлено, как следует утилизировать отходы при выполнении работы. В разделе ЧС были

рассмотрены возможные виды ЧС и меры их профилактики. И в последнем разделе были рассмотрены правовые и организационные моменты социальной ответственности.

### **5.3. Профессиональная социальная безопасность**

#### **5.3.1. Анализ вредных и опасных факторов, которые может создать объект исследования.**

Так как объектом исследования является технология для автоматического управления роботизированными механизмами, анализ вредных или опасных факторов зависит от типа аппарата. Исследование направлено на анализ поведенческих алгоритмов с использованием примитивных моделей, которые не привязаны к конкретным техническим аппаратам, и любые вредные или опасные факторы ограничиваются местом проведения исследования.

#### **5.3.2. Анализ вредных и опасных факторов, которые могут возникнуть на рабочем месте при проведении исследований.**

В данном разделе анализируются вредные и опасные факторы, которые могут возникать при разработке или эксплуатации проектируемого решения. Так как разработка ведётся с применением персонального компьютера, она может спровоцировать следующие вредные факторы:

- Монотонность работы;
- Влажность воздуха.
- Недостаточная освещенность рабочего места;

К опасным факторам при работе за компьютером относятся:

- Статическое электричество;
- Высокое напряжение электрической сети.

Для предотвращения возникновения данных факторов необходимо рассмотреть меры защиты. В противном случае воздействие вредных

факторов может привести к утомляемости и снижению работоспособности, а опасные факторы могут нанести вред работнику – травмы и серьёзные проблемы со здоровьем.

### **Монотонность работы**

Монотонный режим работы заключается в некоторых видах работ, при которых человек долгое время выполняет однообразные элементарные действия либо имеет место быть предельная концентрация внимания на какой-либо деятельности. Работа программиста очень сильно подвержена этому неблагоприятному фактору, так как при написании кода от программиста требуется очень большая концентрация и повышенное внимание. Этот фактор относится к психофизическому типу.

Для борьбы с монотонностью используются следующие методы:

- усложнение рабочих операций, выполняемых действий, объединение их в комплексы;
- увеличение темпа работы или подачи информации (сигналов);
- расчленение общего задания на отдельные части для того, чтобы появились промежуточные (поэтапные) цели;
- прерывание на 5-минутный отдых.

При работе с ПК основным фактором, влияющим на нервную систему специалиста, является огромное количество информации, которое он должен воспринимать. Меры, позволяющие снизить воздействие этого вредного производственного фактора, регулируются СанПиН 1.2.3685-21 [28]. Они позволяют увеличить производительность труда и предотвратить появление профессиональных болезней.

### **Влажность и температура воздуха**

Повышенная или пониженная влажность воздуха также играет значимую роль в процессе работы. Повышенная влажность воздуха создаёт неблагоприятные метеорологические условия – происходит нарушение

терморегуляции и перегревание организма, уменьшается испарение пота, и, следовательно, уменьшается и отдача тепла организмом, что резко ухудшает состояние и работоспособность человека. Низкая относительная влажность воздуха способствует испарению пота, в результате чего происходит быстрая отдача тепла организмом.

Источником отклонений влажности воздуха в помещении могут являться тепловыделяющие приборы (например, ЭВМ и орг. техника), либо охлаждающие устройства (такие как кондиционеры). Для того, чтобы устранить повышенную, либо пониженную влажность, необходимо правильно (в зависимости от требований) настроить систему вентиляции в помещении.

Требования СанПиН 2.2.4.548 – 96 [29] описывают параметры микроклимата в офисах. Они зависят от сезона – холодный или тёплый. Также на данные значения влияет категория физической тяжести работы. Так как во время выполнения работы отсутствуют систематические физические нагрузки, дипломнику присваивается категория 1а. Документ описывает как оптимальные (Таблица 14), так и допустимые (Таблица 15) величины показателей микроклимата.

Таблица 14. Оптимальные значения характеристик микроклимата по СанПиН 2.2.4.548 – 96

Период года	Температура воздуха, °С	Температура поверхности, °С	Относительная влажность, %	Скорость движения воздуха, м/с
Холодный	22-24	21-25	40-60	0,1
Тёплый	23-25	22-26		

Таблица 15. Допустимые величины показателей микроклимата по СанПиН 2.2.4.548 – 96

Период года	Температура воздуха, °С	Температура поверхностей, °С	Относительная влажность, %	Скорость движения воздуха, м/с
Холодный	19-24	18-25	15-75	0,1-0,2
Теплый	20-28	19-29		0,1-0,3

Теперь для проверки соответствия нормам необходимо измерить данные параметры в нашем помещении:

- температура воздуха – 23 градуса Цельсия;
- температура поверхности – 24 градуса Цельсия;
- относительная влажность воздуха – 53%;
- скорость движения воздуха – 0.1 м/с.

Рабочее место оборудовано нижеперечисленными средствами, поддерживающими оптимальный микроклимат:

- центральное отопление;
- вентиляция;
- искусственное кондиционирование.

Рабочее место соответствует нормам по СанПиН 2.2.4.548 – 96.

### **Освещение**

Освещение – получение, распределение и использование световой энергии для обеспечения благоприятных условий видения предметов и объектов.

В рабочем помещении сочетаются естественное освещение (через окна) и искусственное освещение (использование ламп при недостатке естественного освещения). Светильники в помещении располагаются равномерно по площади потолка, тем самым обеспечивая равномерное освещение рабочих мест.

Освещённость на рабочем месте должна соответствовать характеру зрительной работы, который определяется наименьшим размером объекта различения, контрастом объекта с фоном и характеристикой фона. Согласно СНИП 23-05-95 разряд зрительных работ программиста относится к разряду III подразряду Г (высокой точности), параметры искусственного освещения указаны в таблице 16 [30].

Таблица 16. Нормативные значения освещённости

Характер зрительной работы	Разряд зрительной работы	Подразряд зрительной работы	Искусственное освещение		Естественное освещение КЕО ен, % при боковом
			Освещенность при системе общего освещения, лк	Коэффициент пульсации, Кп, %	
Различение объектов высокой точности	III	Г	400	15	1,2

Рассчитаем фактическую освещенность рассматриваемого помещения. Длина и ширина аудитории равны соответственно 6 и 3 м, высота – 3 м, количество светильников  $N = 3$ , количество ламп внутри светильника  $n=2$ . Расположение светильников в кабинете показано на рисунке 28.

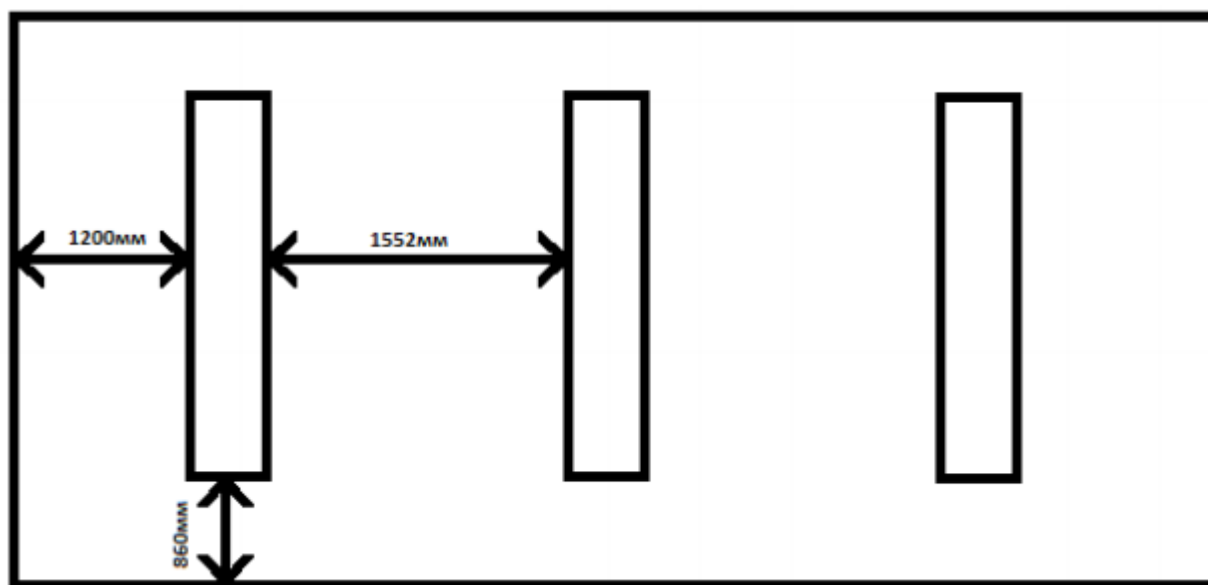


Рисунок 28. Расположение светильников в кабинете

Проведем расчеты индекса помещения по следующей формуле 9:

$$i = \frac{S}{h * (A + B)}; \quad (9)$$

Где  $i$  – индекс помещения;  $S$  – площадь помещения,  $m^2$ ;  $h$  – высота помещения, м;  $A$  – длина помещения, м;  $B$  – ширина помещения.

Подставив приведенные ранее значения переменных, получим:



$$i = \frac{3 * 6}{3 * (3 + 6)} = \frac{18}{27} = 0,66;$$

Так как поверхность стен окрашена в бледно-голубой цвет, с окнами без штор, то коэффициент отражения поверхности стен  $P_{ст} = 50\%$ . Так как поверхность потолка светлая, то коэффициент отражения поверхности потолка  $P_{п} = 30\%$ . Исходя из значения индекса помещения и коэффициентов отражения потолка и стен можно определить, что коэффициент использования рассматриваемого светового светильника с люминесцентными лампами люминесцентными лампами ЛБ-80 (мощность 40 Ватт) равен 38%.

Рассчитаем освещенность помещения по следующей формуле 10:

$$E_{факт} = \frac{N * n * \Phi_{ст} * \eta}{S * K_з * Z}; \quad (10)$$

Где  $E_{н}$  – фактическая освещенность;  $N$  – число светильников в помещении (установлено 3 светильника);  $n$  – число ламп в светильнике (по 2 лампы на светильник);  $\Phi_{ст}$  – величина стандартного светового потока, лм;  $\eta$  – коэффициент использования светового потока (определено ранее 38%, т. е. 0,38);  $S$  – площадь помещения (габариты помещения 3м х 6м);  $K_з$  – коэффициент запаса;  $Z$  – коэффициент неравномерности освещения.

Зная, что  $\Phi_{ст} = 4800$  (лм) для люминесцентных ламп ЛБ-80,  $K_з$  для помещений с малым выделением пыли равен 1,5, а  $Z$  для люминесцентных ламп равен 1,1, рассчитаем значение фактической освещенности по приведенной выше формуле:

$$E_{факт} = \frac{3 * 2 * 4800 * 0,38}{18 * 1,5 * 1,1} = 368.091 \text{ (Лм)};$$

Данное значение не удовлетворяет нормативу. Рассчитаем численную оценку разности между фактическим значением освещенности и нормативным по следующей формуле 11:

$$\Delta E = \frac{(E_{факт} - E_{н})}{E_{н}} * 100\%; \quad (11)$$

Где  $\Delta E$  – показатель разности между фактической освещенностью и нормативной;  $E_{факт}$  – фактическое значение освещенности;  $E_{н}$  – нормативное

значение освещенности.

Подставив значения в формулу, вычислим  $\Delta E \approx 7\%$ . Допустимый показатель отклонения составляет 20%, таким образом, можно сделать вывод о том, что фактическая освещенность находится в пределах допустимой нормы.

### **Статическое электричество**

Статическое электричество может образовываться при эксплуатации электроустановок высокого напряжения постоянного тока и электризации диэлектрических материалов. Предельно допустимый уровень напряженности электростатических полей (ЕПРЕД) устанавливается равным 60 кВ/м в течение 1 ч. При напряженности электростатических полей менее 20 кВ/м время пребывания в электростатических полях не регламентируется [31].

Основными средствами против накопления статики в нашем случае являются:

- отвод зарядов путем заземления оборудования и коммуникаций;
- применение антистатических веществ.

### **Высокое напряжение электрической сети**

Высокое напряжение электрической сети относится к опасным факторам на рабочем месте.

В процессе разработки алгоритма обучения нейронной сети основные работы проводятся с помощью персональных компьютеров, которые могут быть источником поражения электрическим током.

Основные подверженные опасности процессы:

- риск возникновения короткого замыкания, в системном блоке либо периферийном оборудовании;
- непосредственное соприкосновение с оборудованием и внешним окружением помещения, оказавшимся под напряжением;

- поражение электрическим током в результате повреждения изоляции.

Сама рабочая зона сконструирована таким образом, чтобы исключить возможность перетирания заизолированных проводов и контакта внешнего оборудования с возможным источником замыкания. Так же в электросхеме помещения имеются автоматы, отключающие питание при возникновении короткого замыкания. В электрической схеме имеются коллективные способы электрозащиты такие как: изоляция токопроводящих частей, защитное заземление оборудования.

Работы по разработке алгоритма обучения нейронной сети проходят в помещении с нормальными условиями, где влажность не превышает 50%, температура воздуха не более 27 градусов Цельсия, мебель не токопроводящая, и имеются изолированные полы с антистатическим покрытием. Помещение, в котором проходят работы, относится к категории помещений с повышенной опасностью (ПУЭ [33]), так как в условиях производства используется значительное количество электроустановок.

В помещениях с повышенной опасностью для питания ЭВМ, светильников (в частности, светильников с люминесцентными лампами) должно применяться напряжение не выше 220 В переменного или постоянного тока. Защитными мерами являются зануление металлических корпусов светильников. Питание электроприемников должно выполняться от сети 380/220 В с системой заземления TN-S или TN-C-S. В зданиях следует применять кабели с медными жилами, если их расчетное сечение не превышает 16 мм<sup>2</sup> [33].

Защитные меры электробезопасности зданий выполняются в соответствии с требованиями ПУЭ. Во всех помещениях присоединены открытые проводящие части ЭВМ, светильников общего освещения к нулевому защитному проводнику. Подвесные светильники, не оснащенные зажимами для присоединения к нулевому защитному проводнику, отсутствуют. В групповой сети, питающей штепсельные розетки для

переносных электрических приборов, обеспечена дополнительная защита при случайном непреднамеренном прикосновении к опасным токоведущим частям посредством устройства защитного отключения. В качестве дифференциальных автоматических выключателей использованы дифференциальные выключатели, не имеющие источника питания, зависимо от напряжения защищаемой сети. В здании используется безопасная система заземления TN-S (нулевой рабочий и нулевой защитный проводники работают отдельно по всей системе) на ТП-6 кВ через РУ -0,4 кВ, соединенной со зданием пятижильным проводником трехфазной сети.

## **5.4. Экологическая безопасность**

### **5.4.1. Анализ влияния объекта исследования на окружающую среду**

Так как объектом исследования является технология по обучению нейронных сетей управлению роботизированными механизмами, анализ влияния на окружающую среду зависит от типа рассматриваемого механизма. Исследование направлено на анализ поведенческих алгоритмов с использованием примитивных моделей, которые не привязаны к конкретным техническим аппаратам, и любые вредные для экологии факторы ограничиваются процессом проведения исследования.

### **5.4.2. Анализ влияния процесса исследования на окружающую среду**

Работы по реализации проекта не оказывают значительного влияния на окружающую среду, так как в процессе работы не используются вредные химические соединения в концентрациях, превышающих допустимые.

Основные устройства, используемые в работе, это ЭВМ и принтер, что ведет к чрезмерному потреблению энергии и различных расходных материалов: картриджи, бумага.

Из строя может выходить используемая компьютерная техника, что при неправильной утилизации может загрязнять литосферу. В таком случае она утилизируется с уничтожением информации согласно ГОСТ Р 50739-95 [34].

Бумага является самым распространенным побочным продуктом в виде разных документов и чертежей. Снизить потребление и, соответственно, отходы бумаги можно с помощью перехода с бумажных на цифровые носители, введением цифрового документооборота и систем САПР. Также существуют программы по утилизации и переработке макулатуры, согласно ГОСТ Р 53692-2009 [32] в новый бумажный продукт, что снижает потребность в вырубке лесов. При правильной переработке более 75% макулатуры попадает во вторичный оборот.

Картриджи необходимо сдать в организацию, занимающуюся их утилизацией. Утилизация картриджей проводится двумя способами:

- Наиболее распространенный способ – переработка с последующим использованием для новых изделий. Многие компании-производители собранные отработанные изделия передают дочерним перерабатывающим организациям.
- Другой способ – механическое и термическое разложение. Изделия подвергаются сортировке и разделению на составляющие компоненты, после чего пластиковые и металлические части измельчают и переплавляют при высокой температуре с получением вторсырья.

Оба способа выполняются согласно ГОСТ 17.4.3.04-85 Охрана природы (ССОП). Почвы. Общие требования к контролю и охране от загрязнения [35].

Так же в ходе работы может выйти из строя люминесцентная лампа, которая является опасным отходом, так как при нарушении целостности корпуса выделяться пары ртути. Методы безопасной утилизации данного

продукта регламентирована по ГОСТ 12.3.031-83, и в случае поломки будут сданы в пункт приёма отработанных люминесцентных ламп [36]. В г. Томске таких пункта два и оба принадлежат компании "Экотом".

Чтобы решить проблему чрезмерного использования разных расходных материалов (картриджей, бумаги), для предварительных проверок можно использовать бумагу с двух сторон.

## **5.5. Безопасность в чрезвычайных ситуациях**

### **5.5.1. Анализ вероятных ЧС, которые может инициировать объект исследований**

Так как объектом исследования является технология по обучению нейронных сетей управлению роботизированными механизмами, анализ вероятных ЧС, которые может инициировать объект исследований зависит от типа рассматриваемого механизма. Исследование направлено на анализ поведенческих алгоритмов с использованием примитивных моделей, которые не привязаны к конкретным техническим аппаратам, и любые возможные ЧС зависят исключительно от исследовательского рабочего места.

### **5.5.2. Анализ вероятных ЧС, которые могут возникнуть на рабочем месте при проведении исследований**

Наиболее вероятной чрезвычайной ситуацией, которая может возникнуть в офисе во время разработки проекта, является пожар. Его могут вызвать следующие причины:

- несоблюдение мер пожаробезопасности;
- обрыв проводов;
- замыкание электропроводки оборудования.

Комплекс мероприятий, описанных в ГОСТ 12.1.004-91, позволяют уменьшить вероятность возникновения пожара и более оперативно ликвидировать последствия [37].

- регулярная проверка систем оповещения и тушения;
- отключение оборудования при покидании рабочего места;
- проведение инструктажа работников по действиям при пожаре;
- проведение учебной тревоги два раза в год;
- установка систем противопожарной сигнализации;
- закупка огнетушителей;
- оборудование запасных выходов при пожаре;
- создание плана эвакуации и размещение его экземпляров в доступных местах.

Пожарная безопасность в офисном помещении обеспечивается системой пожарной сигнализации. Имеется план эвакуации (Рис.2), закреплен ответственный за пожарную безопасность.



Рисунок 29. План эвакуации при пожаре и других ЧС

В ходе осмотра рабочего места были обнаружены системы противопожарной сигнализации, реагирующие на дым и другие продукты горения. Были найдены установленные огнетушители. На стенах присутствовали планы эвакуации, на которых были указаны запасные выходы.

## **5.6. Выводы по разделу**

При анализе данного раздела были рассмотрены вопросы соблюдения прав персонала на труд, выполнения требований к безопасности и гигиене труда, к промышленной безопасности, охране окружающей среды и ресурсосбережению.

Были проанализированы и проклассифицированы все возможные вредные производственные факторы. Анализируя заявленный проект можно сделать вывод, что проект полностью удовлетворяет нормативным данным.



## Заключение

В настоящей выпускной квалификационной работе исследованы возможности использования нейронных сетей для управления автономными аппаратами. Рассмотрены особенности систем управления автономными аппаратами, изучены различные способы восприятия окружающей среды. Рассмотрена архитектура гибридной системы управления, которая позволяет использовать обученных в смоделированных средах интеллектуальных агентов для управления роботизированными механизмами в реальном мире.

Была спроектирована и разработана виртуальная среда по моделированию работы манипулятора, и обучению интеллектуальных агентов управлению спроектированным устройством. Проведено поэтапное обучение интеллектуальных агентов выполнению задачи по захвату и поднятию объектов. Проведено исследование влияния различных гиперпараметров функции ценности и применения разного количества датчиков восприятия окружения на скорость и качество обучения интеллектуальных агентов. В результате полученных данных была обучена ИНС для управления приводами манипулятора в смоделированной среде. Обученный агент успешно выполняет задачу по быстрому захвату и поднятию целевого объекта в 100% проведённых симуляций.

На основе полученных знаний можно проектировать более сложные системы, моделирующие реальные роботизированные устройства с целью обучения, последующей калибровкой и использованием во внешнем мире.

Перспективами развития изучаемого решения является замена программируемых поведенческих алгоритмов роботизированных устройств на обучаемые. Данная концепция может существенно снизить стоимость разработки ИИ, и многократно расширить спектр задач для автономных аппаратов.

## Список литературы

1. Researchers ran a simulator to teach this robot dog to roll over. [Электронный ресурс] / Интернет-издание о стартапах, интернет-бизнесе, инновациях и веб-сайтах – URL: <https://techcrunch.com/2019/01/17/researchers-ran-a-simulator-to-teach-this-robot-dog-to-roll-over/> (дата обращения 01.03.2021)
2. Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.
3. Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End to-end training of deep visuomotor policies. *Journal of Machine Learning Research*, 17(39):1–40, 2016.
4. Stephen James and Edward Johns. 3d simulation for robot arm control with deep q-learning. *arXiv preprint arXiv:1609.03759*, 2016.
5. Benjamin Planche, Ziyan Wu, Kai Ma, Shanhui Sun, Stefan Kluckner, Terrence Chen, Andreas Hutter, Sergey Zakharov, Harald Kosch, and Jan Ernst. Depthsynth: Real-time realistic synthetic data generation from cad models for 2.5 d recognition. *arXiv preprint arXiv:1702.08558*, 2017.
6. J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, “Domain Randomization for Transferring Deep Neural Networks from Simulation to the Real World,” *arxiv:1703.06907*, 2017.
7. G. Griffin, A. Holub, and P. Perona. The Caltech-256. Technical report, California Institute of Technology, 2007.
8. M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2010 (VOC2010) Results. <http://www.pascal-network.org/challenges/VOC/voc2010/workshop/index.html>.

9. Microsoft Kinect. // Статья – 2021. [Электронный ресурс]. URL:<http://www.xbox.com/en-us/kinect>. (дата обращения 01.05.2021)
10. Jie Tang, Stephen Miller, Arjun Singh, and Pieter Abbeel. A textured object recognition pipeline for color and depth image data. In Robotics and Automation (ICRA), 2012 IEEE International Conference on, pages 3467–3474. IEEE, 2012
11. Alvaro Collet, Dmitry Berenson, Siddhartha S Srinivasa, and Dave Ferguson. Object recognition and full pose registration from a single image for robotic manipulation. In Robotics and Automation, 2009. ICRA'09. IEEE International Conference on, pages 48–55. IEEE, 2009.
12. Fischler, M.A. and Bolles, R.C., "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography", Communications of the ACM, 1981, 24, (6), pp. 381--395.
13. Lowe, D. G., "Distinctive Image Features from Scale-Invariant Keypoints", International Journal of Computer Vision, 60, 2, pp. 91-110, 2004
14. H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, "SURF: Speeded Up Robust Features", Computer Vision and Image Understanding (CVIU), Vol.110, No. 3, pp. 346–359, 2008.
15. Lepetit, V., Fua, P.: Monocular model-based 3D tracking of rigid objects. Foundations and Trends in Computer Graphics and Vision 1(1), 1–89 (Jan 2005)
16. Alvaro Collet, Manuel Martinez, and Siddhartha S Srinivasa. The moped framework: Object recognition and pose estimation for manipulation. The International Journal of Robotics Research, 30(10):1284– 1306, 2011.
17. Гаврилов А.В., Губарев В.В., Джо К.Х., Ли Х.Х., Архитектура гибридной системы управления мобильного робота/ Научный вестник НГТУ, 2004, №2

18. Хайкин, Нейронные сети: полный курс. 2 издание. С. Хайкин. – Москва: Издательский дом "Вильямс", 2006. – 1104 с.
19. Sutton R. S., Barto A. G. Reinforcement learning: An introduction. – Cambridge : MITpress, 1998. – Т. 1. – №. 1.
20. Wiering M., Van Otterlo M. Reinforcement learning //Adaptation, Learning, and Optimization. – 2012. – Т. 12.
21. Universal robots // Статья – 2021. [Электронный ресурс]. URL: <https://www.universal-robots.com/> (дата обращения 01.05.2021)
22. 38 Роботов будущего: обзор полуфиналистов \$10M ANA Avatar XPRIZE // Статья – 2021. [Электронный ресурс]. URL: <https://habr.com/ru/post/558514/> (дата обращения 01.05.2021)
23. Производственный календарь на 2021 год// Статья – 2021. [Электронный ресурс]. URL: <http://www.consultant.ru/law/ref/calendar/proizvodstvennye/2021/> [Дата обращения: 27.05.21]
24. Трудовой кодекс Российской Федерации (ТК РФ).
25. ТОИ Р-45-084-01. Типовая инструкция по охране труда при работе на персональном компьютере.
26. ГОСТ 12.2.032-78 ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования. – М. : Изд-во стандартов.
27. ГОСТ Р ИСО 9241-4-2009. Эргономические требования к проведению офисных работ с использованием видеодисплейных терминалов (VDT). Часть 4. Требования к клавиатуре. – М. : Изд-во стандартов.
28. СанПиН 1.2.3685-21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания»

29. СанПиН 2.2.4.548-96. Гигиенические требования к микроклимату производственных помещений.
30. СНиП 23-05-95 «Естественное и искусственное освещение».
31. СанПиН 2.2.4.1191-03 «Электромагнитные поля в производственных условиях»
32. ГОСТ Р 53692-2009. Ресурсосбережение. Обращение с отходами. Этапы технологического цикла отходов.
33. ГОСТ 12.1.038-82 ССБТ. Электробезопасность. Предельно допустимые уровни напряжений прикосновения и токов. – М. : Изд-во стандартов.
34. ГОСТ Р 50739-95 «Средства вычислительной техники. Защита от несанкционированного доступа к информации».
35. ГОСТ 17.4.3.04-85. Охрана природы (ССОП). Почвы. Общие требования к контролю и охране от загрязнения.
36. ГОСТ 12.3.031-83 «Работы со ртутью. Требования безопасности»
37. ГОСТ 12.1.004-91 «Система стандартов безопасности труда (ССБТ)»

## Приложение А

### Раздел №1

«Development of an algorithm for automatic control of robotic mechanisms using artificial neural networks»

Студент:

Группа	ФИО	Подпись	Дата
8BM93	Пантюхин Евгений Игорьевич		

Руководитель ВКР:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
доцент ОИТ	Иванова Ю.А.	к.т.н.		

Консультант – лингвист отделения иностранных языков, школы ИШИТР:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
ст. преподаватель	Розанова Я.В.			

## Introduction

In today's world, unmanned robotic mechanisms perform a huge range of tasks that can be routine, dirty, dangerous, or simply too expensive using traditional methods. Many tasks can run in parallel, so deploying multiple standalone devices is a natural step. A group of manipulators can effectively perform a wide class of tasks, such as sorting, transportation, assistance in production, in laboratories or at home, work in hazardous places, etc. However, this requires highly accurate autonomy of devices, and traditional methods are ineffective in a wide range of tasks. To improve efficiency, in a number of such tasks, you can use artificial neural networks [1].

Artificial neural networks, which have recently become widespread, are a tool that can effectively solve a very wide range of problems. In particular, these are image and speech recognition, time series forecasting, text translation, diagnosis in hospitals, behavioral algorithms in games, etc. This list grows every year. ANNs, being more efficient, are replacing traditional algorithms. Robotics is no exception.

The aim of this work is to develop an algorithm for automatic control of robotic mechanisms using artificial neural networks.

To achieve this goal, the following tasks were set:

- 1) Study the structure of the control system for unmanned mechanisms
- 2) Explore the existing methods of training neural networks
- 3) Explore the features of teaching ANN to control robotic mechanisms
- 4) Develop software for simulating the operation of a manipulator based on

ANN

## **1. The structure of the control system for robotic unmanned mechanisms**

This section discusses the features of control systems for unmanned mechanisms, various ways of perceiving the environment, methods of teaching behavioral algorithms, as well as the architecture of a hybrid control system.

### **1.1. Unmanned mechanisms control system**

The control system for an unmanned mechanisms generally performs the following functions:

- 1) perception and recognition of incoming information;
- 2) interaction with the operator (human);
- 3) planning and replanning behavior;
- 4) management of the execution of actions;
- 5) drive control;
- 6) communication with other robots.

Which of the listed functions are used in a particular device depends on the device itself. The use of ANN adds one more important function - the ability and the need to learn.

To use neural networks to control an unmanned mechanism, it is necessary to solve several natural problems: minimizing input data and collecting a training sample.

The first problem is the huge flow of incoming data. The use of unmanned mechanisms always implies that the device will interact with the outside world. The best way of perception is visual, but processing visual information using neural networks is an extremely resource-intensive task.

The second problem is the collection of the training sample. For complex or new systems, there is no pre-collected information, and collecting it is a very time-consuming process.



## **1.2. Training algorithms in a simulated environment**

Collecting training data to process information from environmental sensing sensors, or to train behavioral algorithms, in a simulated environment with realistic physics, can significantly increase the impact of machine learning on robotics, providing a faster, scalable and inexpensive way to collect data than is possible with physical robots.

Particularly promising are deep reinforcement learning techniques, with recent results reaching human performance levels in tasks such as Atari [2] and robot control [3]. This requires hundreds of thousands or millions of samples [2], which can take thousands of hours to collect, making it impractical for many devices and applications. However, you need to understand that deep reinforcement learning uses random exploration, which can be dangerous to physical equipment. However, recent research suggests that it is possible to study policies that fully encode complex behavior in simulation, and then successfully run these policies on physical robots with minimal additional training.

Unfortunately, the discrepancies between physics simulators and the real world make it difficult to transfer behaviors from simulation. The process of adjusting simulation parameters to match the behavior of the physical system is time consuming and error prone. Even in a well-designed simulated system, some physical effects are missing, such as flexibility, backlash, wear, etc., which are not captured by modern physics simulators. In addition, low-fidelity imitation sensors such as cameras are often unable to reproduce the realism and noise found in real devices. These differences, collectively known as “reality gap”, create an obstacle to using simulated data on real robots.

One approach to bridging the discrepancy between simulated data and real data is to make the simulator match the physical reality exactly using high quality rendering. The use of only realistic RGB rendering has been investigated in [4], and has had limited success. Using realistic depth data from 3D models through comprehensive modeling of factors such as sensor noise, material reflectance, and

surface geometry allowed models trained on rendered images to translate reasonably well into the real world [5].

### 1.3. Randomizing simulated data

Another promising approach to bridging the discrepancy between simulated data and real data is that instead of collecting data for training in one simulated environment, we randomize the simulator to represent the model in a wide range of environments during training. The result of this procedure, given sufficient variability in the simulation, will be a model that effectively extracts the most important features both in simulation and on real equipment without additional training.

The goal of simulation randomization is to provide sufficient simulated variability during training so that during testing the model can be generalized to real data. An example of this procedure is shown in Figure 1. On the left are scenes modeled with various parameters, on the basis of which artificial neural networks are trained. Shown on the right is a real mock-up for testing trained networks [6].



Fig. 1. Illustration of a randomization method data modeling [6]

The following aspects of the domain can be used in the randomization process:

- shape and size of objects;
- the number of distracting objects;
- position of objects;
- textures of objects;
- textures of the ground, walls, background;
- position, orientation and field of view of the camera;
- number of light sources;
- position, orientation and characteristics of light sources;
- the type and amount of random noise added to the sensors.

The result of this procedure, with sufficient variability in the simulation, will be a model trained in the simulator, and having sufficient generalized experience to work in the real world without additional training.

Domain randomization is a promising area of research aimed at bridging the gap between reality and developing algorithms for the behavior of robots studied in the process of modeling. Deep reinforcement learning can enable more sophisticated simulation behaviors to be learned through large-scale exploration and optimization, and domain randomization can be an important tool for using such policies on real robots.

#### **1.4. Classification and localization of objects**

Object recognition in unstructured scenes is a complex area of modern computer vision research. One important area is robotics, where the ability to quickly and accurately locate objects of interest is critical for versatile robots performing tasks in an unstructured everyday environment such as a home, office or warehouse.

The specific perception problem for robotics has a number of unique features that distinguish it from other computer vision problems. A conventional object recognition system has to deal with a huge number of different objects. One way to deal with this is to introduce a hierarchy by performing recognition at the category level rather than at the instance level. The task of generalizing from a few instances to an entire category of objects remains challenging, and numerous tests and tasks, such as Caltech 256 [7] and PASCAL VOC [8], exist to facilitate progress in this area. On the other hand, for a specific robot in a specific environment, the number of unique objects is relatively small (possibly on the order of hundreds). This allows us to view the problem as an instance recognition problem, collecting a large amount of training data for each object. In addition, the robot can use data from various sensor devices such as cameras and depth sensors.

Object localization in robotics also presents additional challenges that are not present in category-level object recognition tests. Real world environments are very cluttered, contain a lot of overlap, and often contain more different objects in the same scene. Robots often have to interact with some objects in their environment, bypassing others. This means that the robotic sensing system must accurately localize objects once they are detected. In addition, for a robot to respond quickly to changes in its environment, the robotic sensing system must operate in near real time.

#### **1.4.1. Localization and classification using depth maps**

Currently, there are a number of different approaches to the recognition and localization of objects in complex unstructured scenes. The first is the use of sensors to collect color and depth images. Examples of these sensors are various lidars, or simpler sensors of the Kinect type [9], consisting of an infrared projector combined with a monochrome CMOS matrix, which allows obtaining a three-dimensional image in any natural light (Fig. 2).

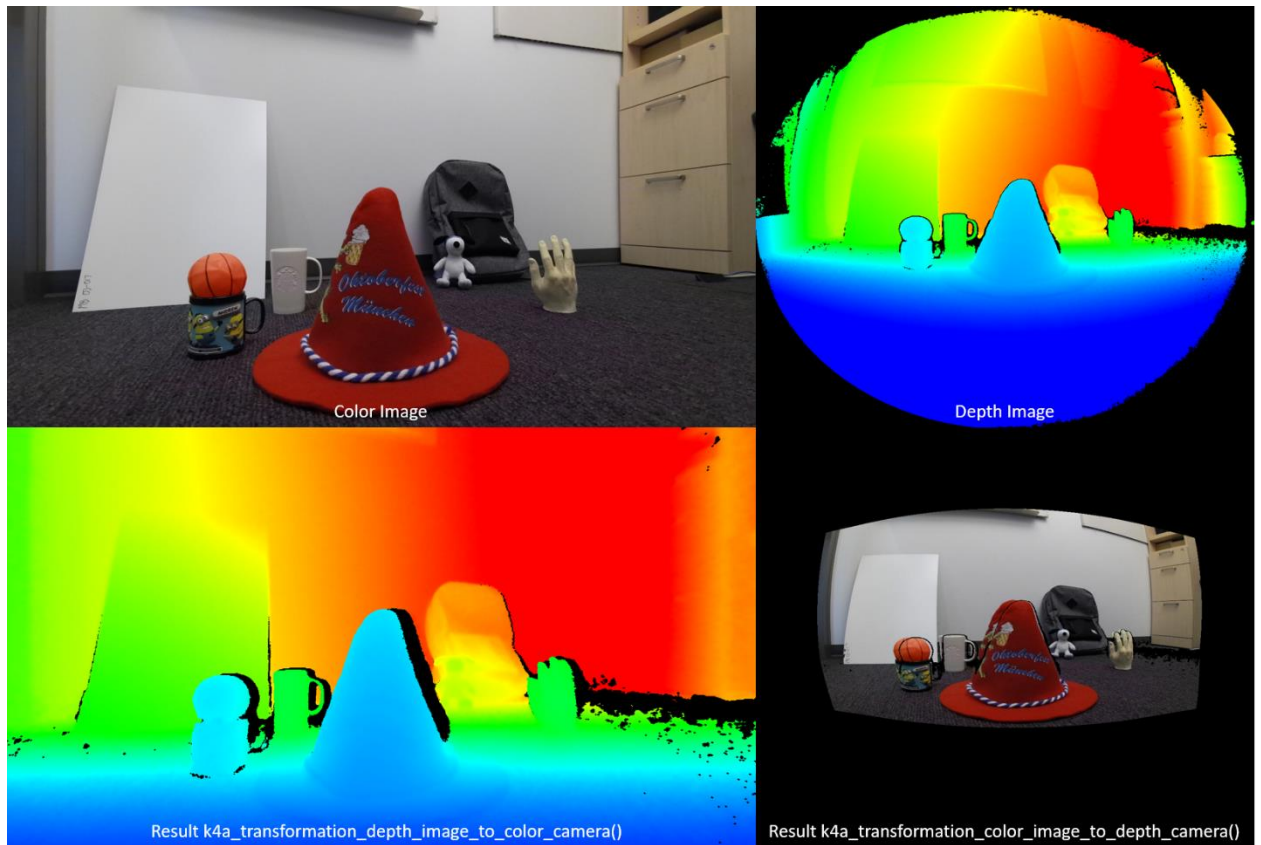


Fig. 2. Received data from the Kinect device

During training to recognize and localize objects based on point clouds, it is possible to more accurately isolate the contours of each object. Based on the contours, color images and depth, a complete point cloud is built, from which three-dimensional mesh models of objects are extracted. Then, from the obtained models, local features of objects are extracted and registered [10].

During testing, taking into account the new color image and point cloud, the scene is segmented into separate objects, choosing a supporting plane using depth information. For each segmented object, classification is performed based on the extracted global and local features obtained at the training stage. After that, various checks of the consistency of the scene with the detected objects take place in order to eliminate classification errors, excluding hypotheses with geometrically incompatible objects, and to refine the segmentation of objects [10].

This algorithm won first place in the Solutions in Perception Challenge, which was held by Willow Garage at ICRA 2011 [11].

### **1.4.2. Localization and classification without using depth maps**

The problem of assessing the positioning of an object model from one image is a well-studied problem in the literature. In the case of point functions, this is known as the Perspective-n-Point (PnP) problem [12]. Assuming that you know a reasonably close match between the 2D image features and the 3D model characteristics, you only need to use the chosen PnP solver to get an estimate of the object's positioning.

The problem of recognizing one object and determining its position from one image requires solving two subproblems:

- finding a sufficiently correct correspondence between the characteristics of the image and the characteristics of the model;
- evaluate the pose of the model that best fits this set of fit.

But even in the presence of well distinguishable invariant features, such as SIFT [13] or SURF [14], mismatches are inevitable, which forces us to use more reliable estimation methods in most modern object recognition systems. A comprehensive review of model-based recognition and tracking methods for 3D objects is available in [15].

The problem of recognizing 3D objects based on models has appeared for a long time. An extensive literature has been written on this topic, but it is far from being solved in its most general form. There are two problems that greatly affect the performance of any recognition algorithm.

The first problem is the complexity of the scene. This problem occurs due to the number of features, and both extreme values significantly reduce the recognition speed. Related to this is the problem of duplicate objects: the ambiguity in the mapping introduced by duplicate object instances is a huge problem for reliable evaluators because the matched features can belong to different object instances, even though they are correct. Solutions such as interpretation trees or image space

clustering are often used, but false positives often occur because algorithms cannot cope with the unexpected complexity of a scene.

The second issue relates to scalability and system latency. In systems operating online, it is necessary to reach a trade-off between recognition accuracy and latency, depending on the requirements for each specific task. In robotics, the response time of robots working in a dynamic environment is often limited by the delay in their perception. Increasing the amount of input data for processing, such as increasing the resolution of a camera, or using multiple cameras, usually leads to serious penalties in terms of processing time.

In general, the algorithm for classifying objects and calculating their exact position relative to the camera is divided into several stages. The first is training, which creates sparse three-dimensional models of objects. For this, a set of images is made with the object in different poses. Further, reliable features are extracted, for example, SIFT [13] or SURF [14], which form a sparse three-dimensional model of the object. Next comes the calibration stage: the correct alignment and scale for each model are calculated to match the actual dimensions of the object and the corresponding coordinate system is determined.

It should be noted that this approach is highly dependent on good calibration of the models, especially in terms of scaling, since the depth is estimated only based on the scale of each model. Therefore, special care must be taken when creating models to establish the correct scale. For example, a modeling error of 1 mm for an object of 10 cm creates an error in determining the depth of up to 3 cm at a distance of 1 m, which is large enough to cause problems for a robotic arm [16].

The stage of object recognition by local features in images occurs in several stages. The first is to find matches. This stage refers to the exact comparison of the characteristics of the image with the characteristics belonging to a particular object. Next comes the process of evaluating the pose of the object, which will be geometrically consistent with the found matches. After that, based on the scale of

the object in the database and on the image, the exact distance to the object is calculated.

It is important to note that the choice of camera and lens can greatly affect the accuracy of the pose estimation. However, cameras are getting better, cheaper, and smaller. This leads to the fact that the parallel use of several cameras becomes readily available to obtain a view of the scene from different angles and high resolution. This approach is effective because it provides depth estimation, resistance to line-of-sight occlusions, and an increased effective field of view.

In addition, higher resolution can potentially improve the recognition of complex objects and the accuracy of pose estimation algorithms, but often with a large penalty, since the additional resolution often leads to an increase in false positives, as well as severe degradation of conditions, latency and throughput.

In addition to all of the above, the use of multiple cameras allows the reconstruction of captured scenes in a virtual environment (Fig. 3). This can greatly simplify the process of training intelligent agents to interact with surrounding objects.



Fig.3. (Columns 1-3) Localization and classification of objects by image, (Column 4) shows the reconstruction of scene data in a virtual environment.

### 1.5. Hybrid control system architecture

In order to reduce the amount of input data for the system responsible for performing any actions, one can use a hybrid approach to the implementation of



artificial intelligence systems [17]. This method is based on dividing the general control system into logical and associative parts (Figure 4).

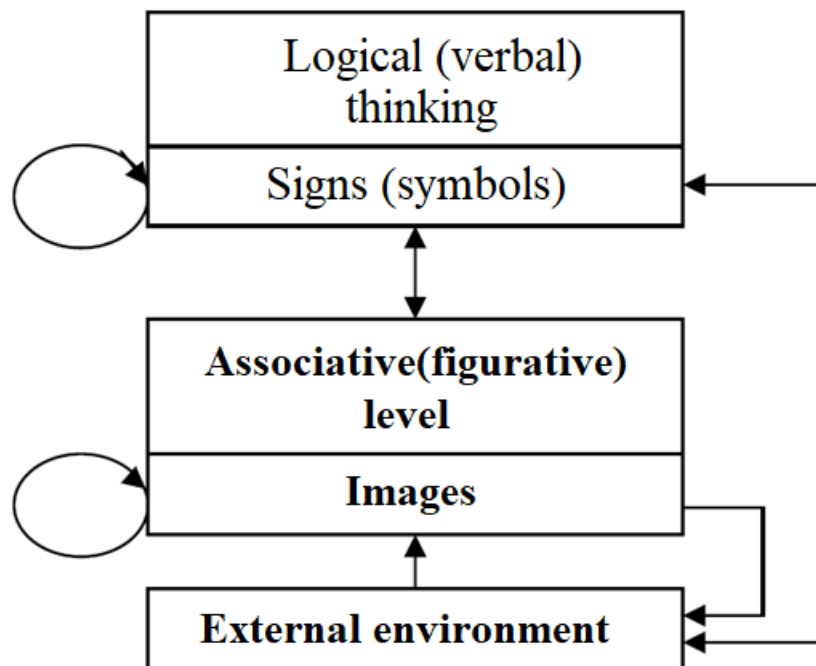


Fig.4. Two levels of intelligent system

In this view, the associative level processes information received from the external environment. Information means any data: image, proximity sensors, oxygen level, temperature, heat sensors, information received from other devices, commands from the operator, etc. All the information received is processed to create associations of various objects and their further classification, clustering and generalization. The set of information depends on the goals that the apparatus must fulfill. Information processing can be fully or partially occupied by ANN.

The logical level processes the information received from the associative level, and on its basis takes any action in the external environment. At this level, both classical engineering methods and artificial neural networks can also be used.

This approach allows dividing the task of controlling an unmanned mechanism into a group of subtasks. This is especially effective when using ANN, since it is much easier to train a group of small neural networks, each of which will perform its own subtask, due to the collection of a training sample.

The most common example of using a hybrid control system is the use of lidars and cameras to detect objects and build an environment at the associative level of information processing, and any algorithms at the logical level (Figure 5). These approaches not only to the section of car control tasks, but also allow testing, and if we are talking about ANN, then training behavioral algorithms on models. The models, in turn, should only simulate the information generated by the associative level, and not recreate a realistic global environment.

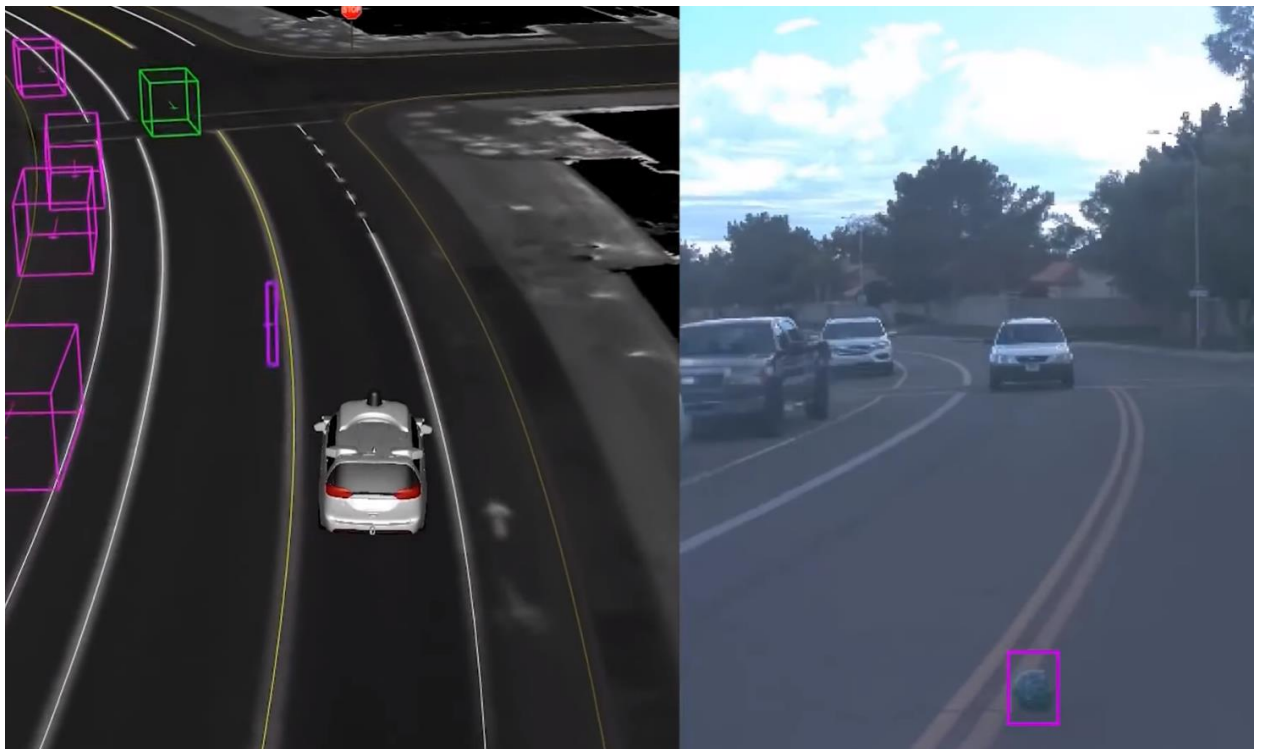


Fig.5. An example of associative transformation of information

This work will consider the possibility of using ANN for processing information at the logical level. Information for training will be collected using a model that simulates similar processes.